

(19) 世界知的所有権機関  
国際事務局



(43) 国際公開日  
2001 年 3 月 8 日 (08.03.2001)

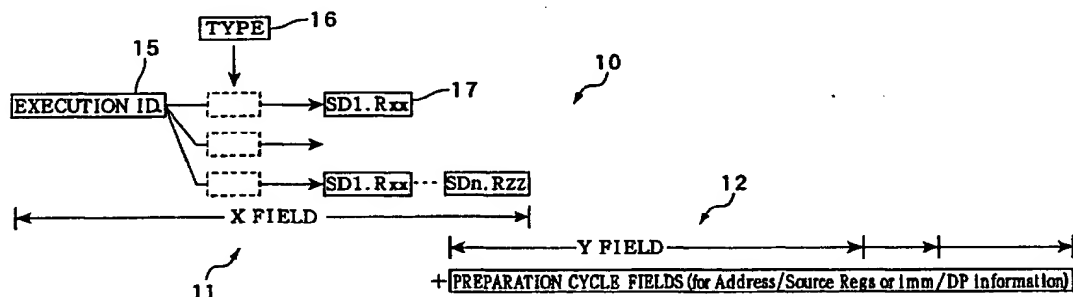
PCT

(10) 国際公開番号  
WO 01/16710 A1

- (51) 国際特許分類<sup>7</sup>: G06F 9/30, 9/38 (74) 代理人: 今井 彰(IMAI, Akira); 〒390-0811 長野県松本市中央1丁目4番20号 日本生命松本駅前ビル8階 Nagano (JP).
- (21) 国際出願番号: PCT/JP00/05848
- (22) 国際出願日: 2000 年 8 月 30 日 (30.08.2000) (81) 指定国 (国内): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ: 特願平11/244137 1999 年 8 月 30 日 (30.08.1999) JP (84) 指定国 (広域): ARIPO 特許 (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), ユーラシア特許 (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI 特許 (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).
- (71) 出願人 (米国を除く全ての指定国について): アイビーフレックス株式会社 (IP FLEX INC.) [JP/JP]; 〒150-0021 東京都渋谷区恵比寿西一丁目16番6号 Tokyo (JP).
- (72) 発明者: 佐藤友美 (SATO, Tomoyoshi) [JP/JP]; 〒305-0046 茨城県つくば市東2丁目18番地10 ルーミつくば 31号202 Ibaraki (JP).
- 添付公開書類:  
— 国際調査報告書
- 2 文字コード及び他の略語については、定期発行される各 PCT ガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

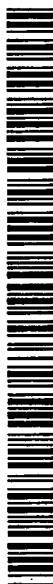
(54) Title: DATA PROCESSOR

(54) 発明の名称: データ処理装置



(57) Abstract: A control program includes an instruction set having a first field where an execution instruction to specify the contents of an operation or another data processing executed by at least one processing unit constituting a data processor can be written and a second field where preparation information, the contents of which is independent of the contents of the execution instruction written in the first field, for setting the processing unit so that an operation or another data processing executed by an execution instruction can be executed can be written. It is possible to prepare execution of succeeding execution instructions thanks to the preparation information. Therefore the problems that cannot be solved by a conventional instruction set, for example, in the case of a branch instruction, the branch destination is written in the second field, and therefore the branch destination can be known previously.

[続葉有]



WO 01/16710 A1



---

(57) 要約:

データ処理装置を構成する少なくとも 1 つの処理ユニットで実行する演算または他のデータ処理の内容を指示する実行命令を記述可能な第 1 のフィールドと、実行命令で実行される演算または他のデータ処理が実行可能な状態に処理ユニットを設定する準備情報を記述可能な第 2 のフィールドとを備えた命令セットを用意し、第 1 のフィールドに記述された実行命令の内容に対し、独立した内容の準備情報が第 2 のフィールドに記述された命令セットを有する制御プログラムを提供可能とする。したがって、準備情報により、後続の実行命令の実行準備が可能となり、たとえば、分岐命令においては、分岐先を第 2 のフィールドに記述することにより事前に分岐先が判るなど、従来の命令セットでは解決できなかった問題を解決できる。

## 明 細 書

## 制御プログラム製品およびデータ処理装置

## 5 技術分野

本発明は、マイクロコードなどにより記述された制御プログラム製品、およびそのプログラムを実行可能なデータ処理装置に関するものである。

## 10 背景技術

汎用的な処理および専用のデジタルデータの処理を行う装置として、マイクロプロセッサ（MPU）、デジタル・シグナルプロセッサ（DSP）といった演算機能を内蔵したプロセッサ（データ処理装置あるいはLSI）が知られている。これらの性能向上に大きく貢献したアーキテクチャ要素として、パイプライン化技術、スーパー・パイプライン化技術、スーパー・スケーラ化技術、VLIW技術、特化型データパス（専用命令）追加を挙げることができる。さらに、分岐予測やレジスタバンク、キャッシュ技術等も挙げることができる。

20 ノン・パイプラインとパイプラインとの性能差は明確である。基本的に同一命令であれば、パイプラインの段数だけスループットは確実に向上する。たとえば、4段のパイプラインでは、スループットは4倍以上となることが期待でき、8段のパイプラインであれば、8倍という計算となり、スーパー・パイプライン技術は、更に性能  
25 を2倍以上向上させたことになる。実際は、プロセスの進歩に従い、クリティカルパスを細分化可能な為、動作周波数の上限を大きく改善するという効果も期待出来るので、その貢献度は更に大きいものとなって現われる。しかしながら、分岐命令のディレイ（ペナルティ）は解消されておらず、スーパー・パイプライン方式のマシン

が成功するか否かは、メモリアクセスや分岐に対応した深い段数の遅延を、コンパイラによる命令スケジューリングなどによってどこまで処理できるかにかかっている。

次に、スーパー・スケーラ技術であるが、これは内部のデータパスを高性能化し、プログラムカウンタ近傍の命令を同時実行するものである。この技術はコンパイラの最適化技術の進歩にも支えられて、4命令から8命令程度の同時実行が可能になったとされている。しかしながら、命令自体はその直前の演算結果やレジスタの結果を頻繁に使用する事が多く、ピーク性能は別にして、フォワーディングや命令再配置、アウト・オブ・オーダー、レジスタリネーミング等の各種テクニックを駆使したとしても同時実行可能な命令数は平均的には上記よりかなり低い値とならざるを得ない。特に、条件分岐命令等を複数実行することは不可能なので、スーパー・スケーラ技術の効果は更に低いものとなる。したがって、プロセッサの性能向上への貢献度としては、平均2.0から2.5倍程度と思われる。非常に相性の良いアプリケーションが仮にあったとしても、現実的な貢献度は4倍以下と考えられる。

VLIW技術が、次の技術として浮上する。これは、予めデータパスを並列実行可能なように構成しておき、コンパイラがこの並列実行を高めるように、最適化を行い目的のVLIW命令コードを生成するという考え方であり、極めて合理的な考え方を採用している。これにより、スーパー・スケーラのように1つ1つの命令の並列実行の可能性をチェックする回路が不要なので、並列実行を行うハードウェアの実装手段としては、極めて有望とされているものである。しかしながら、条件分岐命令などを複数実行できないことは上記と同様であり、そのため、実際の性能に対する貢献度としては、3.5倍～5倍程度と考えられる。しかしながら、画像処理や特殊データ処理を必要とするアプリケーションの処理を用途とするプロセッサを考えると、VLIWも最適な解決策とはならない。特に演算結

果の連続処理を要求されるような用途では、汎用レジスタにデータを抱えながらの演算やデータ処理には限界があるからである。これは従来のパイプライン技術でも同様である。

一方、各種のマトリックス計算やベクトル計算等は、専用回路によりこれを実現した方が高い性能を得られることは過去の経験から良く知られている。このため、現在、世界最高性能を目指す最先端の実装技術では、V L I Wをベースにアプリケーション目的に応じて、各種の専用演算回路を実装して、最高性能を目指すという考え方が主流になりつつある。

しかしながら、V L I Wは、プログラムカウンタ近傍の並列処理実行効率を改善する技術であり、例えば2つ以上のオブジェクトを同時に実行したり、2つ以上の関数を実行するにはあまり有効な手段とはならない。また、各種の専用演算回路を実装することはハードウェアが増加することとなり、その一方で、ソフトウェアのフレキシビリティが低下することを意味する。さらに、条件分岐を実行するときに発生するペナルティの問題を本質的に解決し難い。

そこで、本発明においては、これらの従来のプロセッサを高速化する技術と異なった視点から上記の問題を検討し、新たな解決策を提供することを目的としている。すなわち、パイプラインのようにスループットの向上を図ることができると共に、条件分岐を実行する際のペナルティを解決することが可能なシステム、すなわち、制御プログラム製品と、それを実行可能なデータ処理装置およびその制御方法を提供することを目的としている。さらに、複雑なデータ処理であっても、それらのデータ処理に特化した多種多様な専用回路を用いなくても、それぞれのデータ処理をフレキシブルに、そして高速に実行可能な制御プログラム製品と、それを実行可能なデータ処理装置およびその制御方法を提供することも本発明の目的としている。

## 発明の開示

本願の発明者は、上記のような問題がノン・パイプライン技術から今までの技術に用いられている命令セットの制約から上記のような問題が生じていることを見出した。すなわち、プロセッサにおけるデータ処理を規定するプログラム（マイクロコード、アセンブリコード、機械語など）の命令セット（命令フォーマット）は命令操作（実行命令）とその命令を実行する際に使用するレジスタなどの環境またはインタフェースを規定するオペランドとが組み合わせられたニーモニックコードである。したがって、命令セットを見れば、それによって指示されている処理の内容を完全に把握できるが、命令セットをデコードするまで処理の内容については全く判らない。そこで、本発明においては、命令セットの構成方法そのものを大幅に変更することにより、従来技術では対応の難しかった上記の問題を上手く解決し、データ処理装置の性能を飛躍的に向上できるようにしている。

すなわち、本発明においては、データ処理装置を構成する少なくとも1つの処理ユニットで実行する演算または他のデータ処理の内容を指示する実行命令を記述（記載）可能な第1のフィールドと、実行命令で実行する演算または他のデータ処理が実行可能な状態に処理ユニットを設定する準備情報を記述（記載）可能な第2のフィールドとを備えた命令セットを設け、第1のフィールドに記述された実行命令の内容に対し、独立した演算または他のデータ処理の準備情報が第2のフィールドに記述できるようにしている。そして、本発明においては、この命令セットを有する制御プログラム製品あるいは制御プログラム装置を提供する。この制御プログラムは、データ処理装置が読み取り可能な適当な記録媒体に記録して提供でき、また、その制御プログラムを、コンピュータネットワークあるいはその他の通信を介して伝送される伝送媒体に埋め込んで提供できる。

処理ユニットは、データ処理装置を構成する適当な機能的あるいは

はデータパス的に分割可能な単位であり、制御ユニット、算術演算ユニット、さらには、ある程度コンパクトなデータパスを備えてテンプレート的に取り扱い可能な特定のデータパスを具備した処理ユニットあるいはデータフロー処理ユニットなどが含まれる。

5       さらに、本発明のデータ処理装置は、演算または他のデータ処理を実行する少なくとも1つの処理ユニットと、処理ユニットで実行する演算または他のデータ処理の内容を指示する実行命令を記述可能な第1のフィールド、および実行命令で実行する演算または他のデータ処理が実行可能な状態に処理ユニットを設定する準備情報を  
10       記述可能な第2のフィールドとを具備する命令セットをフェッチするユニットと、第1のフィールドの実行命令をデコードし、その実行命令の演算または他のデータ処理が実行できるように予め設定された処理ユニットにより当該演算または他のデータ処理を進める第1の実行制御ユニットと、第2のフィールドの準備情報をデコード  
15       し、第1の実行制御ユニットの実行内容とは独立して処理ユニットの状態を演算または他のデータ処理が実行できるように設定する第2の実行制御ユニットとを有する。

      また、本発明の、演算または他のデータ処理を実行する少なくとも1つの処理ユニットを有するデータ処理装置の制御方法は、上記  
20       の第1のフィールドおよび第2のフィールドとを具備する命令セットをフェッチする工程と、第1のフィールドの実行命令をデコードし、その実行命令の演算または他のデータ処理が実行できるように予め設定された処理ユニットにより当該演算または他のデータ処理を進める第1の制御工程と、この第1の制御工程とは独立して、第  
25       2のフィールドの準備情報をデコードし処理ユニットの状態を演算または他のデータ処理が実行できるように設定する第2の制御工程とを有する。

      本発明にかかる命令セットは、実行命令を記述する第1のフィールドと、この実行命令とは独立し、レジスタの情報およびイミ-

ディエイトなどの準備情報（準備命令）を記述する第2のフィールドとを備えたものである。したがって、算術命令などにおいては、第1のフィールドにADDなどの命令操作が記述され、第2のフィールドにレジスタを特定する命令あるいは情報が記述されるので、  
5 一見、従来のアセンブルコードと同様の命令セットとなる。しかしながら、実行命令と準備情報は独立であり、同じ命令セット内では対応していない。このため、その命令セットでは制御ユニットなどのデータ処理装置の処理ユニットで実行される処理が特定されないという特性を備えている。すなわち、本発明にかかる命令セットは  
10 従来のニーモニックコードとは大きく異なるものである。そして、従来は1つの命令セットの中に記述されていた命令操作とそれに対応するオペランドを個別に、独立して定義できるようにすることにより、従来の命令セットでは実現できない処理を簡単に実行することができる。

15 まず、第2のフィールドに、後続の命令セットの第1のフィールドに記述される実行命令を実行するための準備情報を記述することができる。これにより、実行命令を備えた命令セットが表れる前に、その実行命令を実行するための準備を行うことができる。すなわち、実行命令で実行する演算またはその他のデータ処理が実行可能な状態に処理ユニットを設定することができる。例えば、ある命令セッ  
20 ト（命令フォーマットあるいは命令レコード）の第1のフィールドにデータ処理装置のある制御ユニットに含まれる少なくとも1つの算術論理演算ユニットを操作する命令を記述し、それに先立つ命令セットの第2のフィールドに、その少なくとも1つの算術論理演算  
25 ユニットに用いられるソース側のレジスタあるいはディスティネーション側のレジスタといった算術論理演算ユニットのインタフェースを規定する命令あるいは情報を記述することができる。これにより、実行命令がフェッチされる前に、算術論理演算ユニットのレジスタ情報がデコードされ、レジスタがセットされ、その後にフェッ



チされた実行命令により所定の論理演算が実行され、その結果が指定されたレジスタに保存される。ディスティネーション側のレジスタは実行命令と共に第1のフィールドに記述することも可能である。

したがって、本発明にかかる命令セットにおいても、パイプライン処理と同様にデータ処理を多段階に分けて実行することが可能でありスループットを向上することができる。また、例えば、ADD, R0, R1, #1234Hという命令は、レジスタR1と#01234Hを加算してこれをレジスタR0に格納するという意味になるが、ハードウェア構成上は、前の命令セットの実行サイクルとオーバーラップさせて、ADDという実行命令を実行する1CLK前にレジスタR0と「#01234H」を算術論理演算ユニットである算術加算器ADDが属するデータパスの入力レジスタにリードを実行しておくことと高速実行させる観点からは、都合が良い。つまり、AC特性上は、純粋に算術加算を行うようにできるので、実行周波数特性が向上する。パイプライン処理において、パイプライン段数を増加させて、レジスタファイルからのリードサイクル専用1ステージ消費する設計方針により、この問題をある程度回避することができる。しかしながら、その結果、遅延は確実に増加することになるのに対し、本発明においては遅延を増加させずに問題を解決できる。

そして、本発明の命令セットにおいては、準備情報を実行命令に先立って記述できるので、条件分岐命令などの分岐命令においては、分岐先の情報が実行命令に先立って制御ユニットに与えることができる。すなわち、従来のニーモニックコードでは、命令セットの内容は人間が一目で分かるが、その命令セットが表れるまで処理内容が判らなかつた。これに対し、本発明にかかる命令セットでは、命令セットの内容は一目では分からないが、実行命令が表れる前に、その実行命令に関連する情報が分かる。したがって、実行命令に先立って分岐先が判るので、その分岐先の命令セットをフェッチすることも可能であり、さらには、分岐先の実行命令に先立ってその準

備を行うことも可能となる。

一般に、現在のCPU/DS Pの殆どがパイプライン処理を後段（時間軸が後方）にシフトすることで、処理の高速化を図ることに成功しているが、プログラムの分岐時やCALL/RET実行時には、この問題が表面化する。つまり、先行してフェッチアドレス情報が得られていない為に、本質的にペナルティとなり、原理的にこれを解消することができない。もちろん、分岐予測やディレイディッド・ブランチ、高速ブランチバッファ、或いはDS Pにて採用されている高速ループ処理技術等は、このペナルティをかなり緩和する事に成功しているが、連続分岐が数多く発生したりすると、その問題点が表面化し、本質的な解決にはなっていないことは周知の事実である。

また、後続命令が必要とするレジスタ情報が先に得られない為に、パイプライン処理を高速化する為のフォワーディング処理やバイパス処理の複雑さが増大し、従来技術で高速化を図ろうとすること自体が膨大なハードウェア・コストの上昇を招く要因となる。

したがって、従来の命令セットでは、分岐先のアドレス情報はデコード後にしか得られず、条件分岐を実行するときに発生するペナルティを本質的に解決し難いのに対し、本発明の命令セットにおいては、分岐先の情報を事前に与えることができるので、条件分岐を実行するときのペナルティを無くすことができる。さらに、ハードウェアに余裕があれば、分岐先の準備命令をフェッチして、それに続く実行命令のための準備を行うことも可能となる。分岐条件が整わない場合は、その準備が無駄になるだけであり、実行時間のペナルティになることはない。

また、後続命令が必要とするレジスタ情報が、実行命令と同時に、あるいは先立って判るので、ハードウェア・コストを増大させずに高速化を図ることが可能となる。つまり、本発明において、従来はハードウェア側にて行っていたパイプライン処理の 1 ステージ分の

処理を、コンパイル時やアッセンブル時に、ソフトウェア処理により静的に事前に実現する事に成功している。

本発明のデータ処理装置において、準備情報に基づく処理を実行する第2の実行制御ユニットは、FPGA (Field Programmable Gate Arrays) のようにトランジスタ間の接続を変更可能なアーキテクチャを動的に制御できるものであっても良い。しかしながら、FPGAのようにハードウェアをダイナミックに変更するには時間がかかり、また、その時間を短縮するためのハードウェアが必要となる。FPGAの再構成情報を二面以上のRAMに保持し、バックグラウンドで実行する事により、見かけ上短い時間で動的なアーキテクチャ変更を行う方式も可能であるが、もし、数クロック以内にこの再構成を行う事を可能とするためには、考えられる組み合わせの数の再構成情報を全て格納するRAMを実装する必要がある、これは、本質的にFPGAの再構成時間が大きく掛かるという経済的な問題を一切解決していない。また、FPGAが、本来ハードウェアのゲートに注目したマッピングを効率良く実現しようとするために抱えている問題、即ち実用上のAC特性の悪さをについては、当面解決出来そうも無い。

これに対し、本発明においては、準備情報として、処理ユニットの入力および／または出力インタフェースを、その処理ユニットの実行時期とは独立して、別に規定し、第2の実行制御ユニットあるいは第2の制御工程において、処理ユニットの入力および／または出力インタフェースを、その処理ユニットの実行時期とは独立して、別に設定することが可能となる。このため、複数の処理ユニットを備えたデータ処理装置においては、第2の実行制御ユニットあるいは第2の制御工程において、これらの処理ユニットによるデータパスの組み合わせを制御することが可能となる。すなわち、第2のフィールドに、データ処理装置に含まれる少なくとも1つの算術論理演算ユニットなどの処理ユニットのインタフェースを規定する命

令を記載あるいは記述することにより、データフロー指定を行うことが可能となる。これにより、データパスの独立性を高めることが可能となり結果的にデータフロー指定を別命令プログラムを実行しながら行ったり、アイドル状態にある制御ユニットあるいはデータ  
5 処理装置の内部のデータパスを、外部の他の制御ユニットあるいはデータ処理装置において実行されている緊急度の高い処理のために貸し出すことも許す構造を容易に提供可能である。

さらに、準備情報に、処理ユニットの処理内容または回路構成も規定する情報を採用し、第2の実行制御ユニットまたは第2の制御  
10 工程により、処理ユニットの処理内容または回路構成も規定することによりさらにフレキシブルにデータパスを構成することができる。

また、第2の実行制御ユニットあるいは第2の制御工程に、レジスタ情報をデコードしてフェッチするなどの算術論理演算ユニットのインタフェースや、他の処理ユニットのインタフェースを規定する  
15 スケジューラとしての機能を持たせてデータパスの組み合わせを管理することにより、多種多様なデータ処理に対応することができる。例えば、ある一定時間だけ、マトリックス計算を行い、その後にフィルター処理を行う場合は、予めそれらの処理に必要なデータ処理装置内部の処理ユニット間の接続を指定し、時間を計数するカ  
20 ウンターを使ってこれを実現する事が出来る。計数カウンターを別の比較回路や外部イベント検出器に置き換える事で、より複雑で柔軟性のあるスケジューリング処理を実現可能となる。

個々の処理ユニットにFPGAのアーキテクチャを採用することが可能である。しかしながら、ハードウェアをダイナミックに変更  
25 するには時間がかかり、また、その時間を短縮するためのハードウェアが必要となる。このため、アプリケーションの実行中に処理ユニット内部のハードウェアを動的に制御することは難しい。仮に、これを複数のRAMをバンク構成にして、瞬時に切り換える方式にしたとしても、数クロック～数十クロック単位での切り換えを実現

する為には、相当数のバンク構成が必要となり、基本的にFPGA内部のマクロセル一つ一つが独立してプログラム構成可能な構造にすると同時に、この切り換えタイミングを検出し、プログラムによる制御機構を持たせる必要がある。しかし、このような構成に対処

5 することは現状のFPGAでは不十分である。さらに、対処可能となったとしても、動的に制御するためには切替のタイミングなどを制御するために、本発明にあるような新しい命令制御機構が必要である事を意味する。

このため、本発明においては、処理ユニットとして、特定の内部

10 データパスを備えた回路ユニットを採用することが望ましい。すなわち、ある程度コンパクトなデータパスを備えた処理ユニットをテンプレート的に用意しておき、そのデータパス間の組み合わせを指示してデータフロー型の処理に持ち込むと共に、準備情報あるいは準備命令により、処理ユニットの内部データパスの一部を選択して

15 処理ユニットの処理内容を変更することにより、さらにフレキシブルに、そして短時間にハードウェアを再構成できる。

たとえば、適当な論理ゲートと、この論理ゲートと入出力インタフェースを接続する内部データパスを予め備えたテンプレート的に使用可能な特定のデータパスを備えた処理ユニットは、以下の説明

20 においてはテンプレートと称されている。このような処理ユニットであれば、入出力されるデータの順番を変えたり、論理ゲート間の接続あるいは選択を変えることにより処理ユニットの処理内容を変更できる。そして、トランジスタレベルで回路を再構成するFPGAに比較すると、予め用意された内部データパスの一部を選択する

25 だけで良いので、短時間で処理内容を変更できる。さらに、予め用意された内部データパスを使用するので、冗長な回路要素は少なく、トランジスタの面積利用効率も高い。したがって、実装密度も高く、経済的である。さらに、高速処理に適したデータパスを構築でき、AC特性も高い。このため、本発明においては、準備情報により、

第2の実行制御ユニットおよび第2の制御工程において、処理ユニットの内部データパスの一部を選択可能とすることが望ましい。

さらに、準備情報に基づき設定された各処理ユニットのインタフェースを保持するスケジュールを管理するように、第2の実行制御  
5 制御ユニットは処理ユニットのインタフェースを管理するスケジューラとしての機能を備えていることが望ましい。

また、準備情報により、複数の処理ユニットにより構成される処理ブロックの入力および／または出力インタフェースを規定できるようにすることが望ましい。複数の処理ユニットのインタフェース  
10 を1つ命令で変更可能とすることにより、複数の処理ユニットが関連するデータパスの変更が1命令で処理することができる。したがって、第2の実行制御ユニットあるいは工程では、準備情報に基づき、複数の処理ユニットにより構成される処理ブロックの入力および／または出力インタフェースを変更可能であることが望ましい。

さらに、処理ブロックの入力および／または出力インタフェースを規定する複数のコンフィグレーションデータを格納したメモリを  
15 設け、準備情報によりメモリに格納された複数のコンフィグレーションデータの1つを選択し、処理ブロックの入力および／または出力インタフェースを変更できるようにすることが望ましい。データフロー指定命令によりコンフィグレーションデータを指定できるようにすることにより、命令自体は冗長にせずに複数の処理ユニットのインタフェースの変更をプログラムから制御することができる。

さらに、処理ユニットとして算術論理演算ユニットを備えた汎用  
25 処理に適した第1の制御ユニットと、処理ユニットとして特定のデータパスを具備する複数のデータフロー処理ユニットを備えた専用処理に適した第2の制御ユニットとを設けることにより、ネットワーク処理や画像処理などの高速性およびリアルタイム性が要求される処理に適したシステムLSIを提供することが可能となる。そして、本発明の命令セットであれば、第1のフィールドに、算術論理

演算ユニットを操作する実行命令を記述でき、第2のフィールドに、算術論理演算ユニットおよび／またはデータフロー処理ユニットのインタフェースを規定する準備情報が記述することが可能であり、上記のシステムLSIの制御に適したプログラム製品を提供できる。

5 従来は、複雑なデータ処理は、専用回路を用意し、その専用回路を用いる専用命令化するしか対応方法が無くハードウェアコストが増大する。これに対し、本発明の命令セットにおいては、実行命令とは独立して第2のフィールドにより論理演算ユニットのインターフェースおよびその処理内容を記述できるので、パイプライン制御  
10 やデータパス制御の構造を命令セットの中に取り込むことが可能となる。したがって、本発明は、プログラムカウンタ近傍の並列処理を実行だけでなく、2つ以上オブジェクトの同時擬似実行や2つ以上の関数の同時擬似実行に有効な手段を提供することになる。つまり、従来の命令セットでは、2つ以上のコンテキストの異なるデータ処理やアルゴリズム実行等の、それぞれ離れたプログラムカウンタに基づく処理が同時に起動ができなかったのに対し、本発明の命令セットを用いてデータフローを適当に定義することにより、プログラムカウンタにかかわらずに処理を実行することが可能となる。  
15

したがって、本発明の命令セットを用いると、並列処理に対して、  
20 予めアプリケーション側から見て性能向上に有効と思われるデータパスを第2のフィールドを用いてソフトウェアから組み込むことが可能であり、それにより実現されたデータパス（データフロー）を必要に応じて、さらにソフトウェアから命令レベルで起動することができる。このデータパスは、特定の目的に対応したデータ処理だけでなく、一般のステートマシンを起動するような目的にも使用可能  
25 なので、極めて自由度が高い。

また、この第2のフィールドの情報により、先行して次命令の準備サイクルを簡単に発生させることが可能となるために、従来はその演算対象をレジスタにせざるを得なかったものが、バッファリン

5      グを前提とすればメモリ（シングルポート／デュアルポート）やレジスタファイルで代用可能となる。すなわち、第2のフィールドに、処理ユニットなどに含まれるレジスタまたはバッファとメモリの間の入出力を指示する命令を記述することを可能とし、第2の実行制御ユニットまたは第2の制御工程において、レジスタまたはバッファとメモリの間の入出力を制御する機能を持つようにすれば、実行命令とは独立してメモリに対する入出力を行うことができる。

10      このことは、1つ1つの命令シーケンスの関連性を高めると同時にハードウェアリソースの競合を事前に回避する事に貢献するので、複数命令の並列同時実行や外部からの割り込み要因への対応を早めることが可能となる。そして、基本的に、メモリをレジスタと見なせるので、高速なタスクスイッチの実現が可能となる。さらに、従来のファーストフェッチのペナルティを消せないキャッシュ・メモリの代わりに、プリローディング型の高速バッファを採用する事も  
15      可能となる為、100%のヒット率を保証しながら一切ペナルティの発生しない高速の組み込みシステムの実現も可能となる。

20      すなわち、メモリをレジスタとみなせるようにすることにより、割り込み等の複数の非同期処理要求に対し高速対応が可能となり、複雑なデータ処理や連続データ処理への対応を非常にフレキシブルに行うことができる。また、レジスタの対比および復帰に時間がかからないので、タスクスイッチ等への高速対応が極めて簡単である。そして、外部メモリと内部メモリのアクセススピード差の影響を完全に消すことができるので、キャッシュは、ファーストフェッチ・ペナルティの問題を効率良く解決できるといったメリットを得ることが  
25      できる。したがって、CALL／RETや割り込み処理／IRETを高速で処理することができるので、イベントに対する応答環境を簡単に構築でき、イベントによってデータ処理性能が低下するのを防止できる。

さらに、第1または第2のフィールドを、VLIWのように、複



数の実行命令または準備命令を記述なフィールドとし、第1または第2の実行制御ユニットが第1または第2のフィールドに記述された複数の独立した実行命令または準備命令を独立して処理可能な複数の実行制御部を備えているようにすれば、さらにパフォーマンスを向上できる。

そして、本発明にかかる制御ユニットをコアあるいは周辺回路に採用したデータ処理装置を実現することにより、上述したようなメリットを活かし、処理速度が速く、さらに経済的なデータ処理装置を提供できる。

#### 図面の簡単な説明

図1は、本発明の命令セットの概要を示す図である。

図2は、図1に示す命令セットのYフィールドをさらに詳しく説明する図である。

図3は、図1に示す命令セットを実際に用いた簡単な例を示す図である。

図4は、図3に示す命令セットによりデータがレジスタに格納される様子を示す図である。

図5は、本発明の命令セットを実行可能なデータ処理装置の例を示す図である。

図6は、従来のCPUあるいはDSPで実行可能なサンプルプログラムである。

図7は、本発明にかかるデータ処理装置のプログラム例である。

図8は、図7に示すプログラムを本発明にかかる命令セットの実行プログラムにコンパイルした例を示す図である。

図9は、本発明にかかるデータ処理装置の異なるプログラム例である。

図10は、図9のプログラムにより構成されたデータフローを示す図である。

図 1 1 は、本発明の命令セットによりデータ処理を実行可能なデータ処理装置の概略構成を示す図である。

図 1 2 は、テンプレートの組み合わせを変えて異なる専用回路を構成する様子を示す図である。

5 図 1 3 は、テンプレートの一例を示す図である。

#### 発明を実施するための最良の形態

以下に図面を参照して、本発明をさらに詳しく説明する。図 1 に、本発明にかかる命令セット（命令フォーマット）の構成を示してある。本発明にかかる命令セット（DAP/DNAの命令セット）10は、第1のフィールドである命令実行基本フィールド（Xフィールド）11と呼ばれる部分と、次の命令実行の効率化を図ることができる第2のフィールドである次命令実行準備サイクル（追加フィールドあるいはYフィールド）12と呼ばれる2つのフィールドを備えている。命令実行基本フィールド（Xフィールド）11は、加減演算、論理和、論理積、比較などのデータの演算、および分岐などのその他の各種のデータ処理の内容を指定し、その結果が格納される先（ディスティネーション）を指定する。また、Xフィールド11は、命令長の使用効率を上げるために実際に実行される命令の情報しか含まない。一方、追加フィールド（Yフィールド）12は、同一の命令セットのXフィールド11の実行命令とは独立した命令（情報）が記述可能であり、たとえば、次の命令の実行準備サイクルに割当てられる。

さらに詳しく命令セット10を説明すると、Xフィールド11は、算術論理演算ユニットなどの処理ユニットに対する命令操作あるいは実行命令（Execution ID）を記述する実行命令フィールド15と、Yフィールド12の有効／無効およびYフィールド12で示す準備命令（準備情報）のタイプを示すフィールド（タイプフィールド）16と、ディスティネーションのレジスタを示すフィールド1

7とを備えている。タイプフィールド16の内容は、Yフィールド12に関連したものであり、Xフィールド11の他のフィールドの内容とは独立して定義できることは上述した通りである。

5 また、Yフィールド12は、タイプフィールド16によって規定される準備情報が記述される。このYフィールド12に記述される準備情報は、演算または他のデータ処理を実行可能な状態にするための情報であり、図2に具体的な幾つかの例を示してある。先ず、TYPEフィールド16はXフィールド11に含まれているが、実行命令フィールド15とは独立あるいは無関係に記述できる。そして、Yフィールド12には、アドレスID (AID) 21と、それによって利用目的が規定されるアドレス情報22、たとえば、アドレス (ADRS)、入出力アドレス (ADRS. FROM/TO) などを記述するアドレス情報フィールド26として利用することができる。このYフィールド12に記述されたアドレス情報は、レジスタあるいはバッファとメモリ (レジスタファイルを含む) との間のリードおよびライトに用いられ、DMAのようにブロック転送も可能な構成になっている。さらに、入出力 (R/W) だけでなく、分岐命令を実行したときの分岐先を示すアドレス (フェッチアドレス、F)、並列実行するときのスタートアドレス (D) などの情報もアドレス情報としてYフィールド12に記述することができる。

15 また、レジスタタイプの命令、たとえば、算術演算あるいはその他の論理演算命令 (MOVE、メモリーリード/ライトなども含む) に対してソース側となるレジスタ (Reg) の情報あるいは即値 (イミディエイト、imm) を規定する情報23もYフィールド12に記述することができる。すなわち、Yフィールド12を以降の実行命令のためのソースを規定するフィールド27として利用することができる。

25 さらに、Yフィールド12には、算術論理演算ユニット (ALU) あるいは他のデータ処理ユニット、たとえば所定のデータパスを

備えたテンプレートのインタフェース（ソース、ディスティネーション）および処理内容の組み合わせを規定する情報 25 も記述することが可能である。すなわち、Y フィールド 12 は、リコンフィグラブルなデータパスなどを、特定のデータ処理を行うために、それらのパイプライン（データフローあるいはデータパス）を定義するためのデータフロー指定命令 25 を記述するフィールド 28 として利用することができる。もちろん、Y フィールド 12 には、そのデータフローをスタートする情報および終了する情報を記述することが可能である。したがって、Y フィールド 12 を用いてリコンフィグラブルなデータパスを定義して生成したデータフローによりコード RAM からコードをフェッチするプログラムカウンタとは独立した処理を行うことができる。

なお、図 1 および図 2 に示した命令セットのフォーマットは、本発明にかかる 2 つの独立した命令フィールドを備えた命令セットの一例であり、これに限定されないことはもちろんである。たとえば、X および Y フィールド内でのフィールドの位置は限定されるものではない。また、独立したフィールド、例えば、タイプフィールド 1 6 の位置は、本例に限定される必要はなく、Y フィールド 1 2 の先頭に位置させることも可能である。また、X フィールド 1 1 と Y フィールド 1 2 の順番を変えることも可能である。本例においては、実行命令が記述される X フィールド 1 1 に Y フィールド 1 2 の情報を含ませることにより X フィールド 1 1 をデコードすることで、Y フィールド 1 2 に準備情報があるか否か、およびその情報の種類を判断できるようにしている。

また、以下ではXフィールド11およびYフィールド12に実行命令あるいは準備命令が記載あるいは記述された例を説明するが、これらのフィールドに命令を記述せず（NOPを記述し）、Xフィールド11あるいはYフィールド12だけが意味を持つような命令セットも可能である。さらに、Xフィールド11に記述された実行

命令にかかるレジスタ情報などのオペランドを備えた準備命令、すなわち、同一命令セット10のYフィールド12に、Xフィールド11の実行命令に対し独立していない準備命令が同時に記述された命令セットも可能である。そして、これらの命令セットを、本発明の、Xフィールド11とYフィールド12が独立し、同一命令セット内では無関係となった命令セットと混在してプログラミングすることも可能である。以下では本発明をわかりやすく説明するためにそのような例を具体的には記載していない。しかしながら、Xフィールド11とYフィールド12に記述された内容が独立した命令セット10と、XフィールドとYフィールドに記述された内容が関連した命令セットが混在したプログラム製品あるいはプログラムを記録した記録媒体なども本発明の範囲に含まれる。

図3に、本例の命令セット10の簡単な例を示してある。j-1番目の命令セット10であるT(j-1)は、そのXフィールド11のタイプフィールド16に、同一の命令セットのYフィールド12に32ビットのイミディエイトが記述されていることが示されている。そして、その命令セットT(j-1)のYフィールド12には、イミディエイトとして「#00001234H」が記載されている。次のj番目の命令セットT(j)には、Xフィールド11の実行命令フィールド15にMOVEが記述され、ディスティネーションフィールド17にレジスタR3が記載されている。このため、このj番目の命令セットT(j)をフェッチすると、制御ユニットのALUは、前の命令フィールドT(j-1)に定義されたイミディエイト「#00001234H」をレジスタR3に格納する。

このようにして、本例の命令セット10（以降では、j番目の命令セット10を命令セットT(j)で示す）では、実行命令が記述された命令セットT(j)の前の命令セットT(j-1)によりその実行命令の準備が行われる。したがって、命令セットT(j)だ

けでは制御ユニットを構成するALUが実行する処理内容は判らないが、2つの命令セット $T(j-1)$ および $T(j)$ によりALUが実行する処理内容は一義的に決定される。また、命令セット $T(j-1)$ の実行命令フィールド15には、その命令セットのYフィールド12とは独立して命令セット $T(j-1)$ の前の命令セットのYフィールド12により準備された処理を実行する命令が記述されている。さらに、命令セット $T(j)$ のタイプフィールド16およびYフィールド12には、次の命令セットの実行命令フィールドに記述された実行命令の準備をする情報が記述されている。

本例では、ある実行命令がXフィールド11に記述された命令セット $T(j)$ の直前の命令セット $T(j-1)$ のYフィールド12に、その実行命令の準備情報（準備命令）が記述されている。すなわち、準備命令のレイテンシーが1クロックの例となっているが、準備情報が記述される命令セットは、直前の命令セットにかぎられるものではない。例えば、複数のALUを備えた制御ユニットの制御プログラム、あるいは後述するデータフロー制御を目的とする準備命令などであれば直前の命令セットである必要はない。準備命令によってセットされたALUの状態（環境あるいはインタフェース）あるいはテンプレートの構成が、その準備命令に対応する実行命令を備えた命令セットがフェッチされて実行されるまで保持されるのであれば、実行命令を備えた命令セット10の数命令前の命令セット10のYフィールド12で準備命令を記述できる。

図4に、図3に示した命令セットによりレジスタとして機能するレジスタファイルあるいはメモリに値が格納される様子を示してある。プロセッサが $j-1$ 番目の命令セット $T(j-1)$ をフェッチして、そのYフィールド12の準備命令によりイミディエイト「#00001234H」がプロセッサのALUのソース側のレジスタDPO.Rにラッチされる。そして、プロセッサが次の $j$ 番目の命令セット $T(j)$ をフェッチし、そのXフィールド11の実行命

令であるMOVEを実行するサイクルでバッファ29bにストアされる。その後、メモリまたはレジスタファイル29aのレジスタR3のアドレスにバッファ29bの値が格納される。したがって、格納先がレジスタではなくメモリであっても、本例の命令セット10を用いると、準備情報に基づく処理を実行命令に先立って行うことにより、実行命令のサイクルでデータをロードあるいはストアすることができる。

図5に、本例の命令セット10により処理内容が記述されたプログラムを実行可能な制御ユニット30を備えたプロセッサ（データ処理装置）38の概略構成を示してある。本例の命令セット10を具備したマイクロコードあるいはマイクロプログラム18はコードROM39に記憶されている。制御ユニット30は、コードROM39からマイクロプログラムの命令セット10をプログラムカウンタによって随時フェッチするフェッチユニット31と、フェッチされた命令セット10のXフィールド11をデコードしてALU34の処理内容を決定あるいはアサートすると共に、ALU34の論理演算結果をディスティネーションのレジスタ34dを選択してラッチする機能を備えた第1の実行制御ユニット32を備えている。

さらに、制御ユニット30は、フェッチされた命令セット10のYフィールド12をXフィールド11のタイプフィールド16の情報に基づいてデコードし、演算処理ユニット（ALU）34のソース側のレジスタ34sを選択する機能を備えた第2の実行制御ユニット33を備えている。この第2の実行制御ユニット33は、タイプフィールド16の情報を除き、Yフィールド12の命令あるいは情報をXフィールド11の内容とは独立して解釈することができる。第2の実行制御ユニット33は、さらに、Yフィールド12に記述された情報がデータフローを規定するものであれば、ALU34のソース側およびディスティネーション側の選択あるいは設定、すなわち、ALU34のインタフェースを決定し、さらに、その状

態を所定のクロックあるいは解除の指示があるまで連続的に保持する機能も備えている。また、Yフィールド12の情報がデータフローを規定する場合は、この第2の実行制御ユニット33は、さらに、ALU34の処理内容も決定し、その状態を所定の期間保持する。

したがって、第1の実行制御ユニット32は、Xフィールド11の実行命令をデコードし、その実行命令の演算または他のデータ処理が実行できるように予め設定された処理ユニットにより演算または他のデータ処理を進める第1の制御工程を行う。一方、第2の実行制御ユニット33は、Yフィールド12の準備情報をデコードし、第1の実行制御ユニット32の実行内容、およびこの第1の実行制御ユニット32で行われる第1の制御工程とは独立に、処理ユニットの状態を演算または他のデータ処理が実行できるように設定する第2の制御工程を行う。

本例の制御ユニット30は、さらに、このような実行制御ユニット32および33と、ALU34の組み合わせを複数備えており、これらによって様々な処理が実行できるようになっている。したがって、本例の制御ユニット30をコアあるいは周辺回路として画像データを高速で処理するようなDSP、汎用のデジタル処理を高速で行えるCPUあるいはMPUなどを構成することが可能である。

図6ないし図9に、本例の制御ユニット30で実行するプログラムの一例を示してある。図6に示したサンプルプログラム41は、従来のCPUあるいはDSPで実行可能なように作成した例である。このプログラムは、#STARTのアドレスから始まるテーブルから最も大きな値を抽出し、最終データであることを示す#ENDを検出すると終了するプログラムである。

図7に記載したプログラム42は、図6と同じ処理を本発明にかかる命令セットを実行可能な制御ユニット30に適したプログラムに変換したものであり、2命令を1つの命令セットで実行できる例を示してある。図7に示したプログラムは、コンパイラを通して本



発明にかかる命令セットの実行プログラムに変換され、制御ユニット 30 で実行される。

図 8 にコンパイルされた本発明の命令セット 10 を有するプログラム 43 を示してあり、このような命令セット 10 を有するプログラム製品 18 が ROM 39、RAM あるいは他の適当なデータ処理装置で読取可能な記録媒体に記憶されて提供される。また、ネットワーク環境で交換される伝送媒体にプログラム製品 43 あるいは 18 を埋め込んで流通することも可能である。このプログラム 43 と、プログラム 42 とを比較すると判るように、第 1 の番目の命令セット 10 の Y フィールド 12 で 2 番目の命令セット 10 の実行命令 15 の準備が行われる。すなわち、タイプフィールド 16 に準備情報としてイミディエイトが Y フィールド 12 に記述されていることが示されており、Y フィールド 12 をデコードした第 2 の実行制御ユニット 32 によりイミディエイトが ALU 34 のソースとなるキャッシュあるいはレジスタに提供される。そして、2 番目の命令セット 10 を実行するときは、その実行命令を行う準備が整った ALU 34 に対し実行命令 15 を行うことができる。すなわち、ディステーションフィールド 17 に規定されたレジスタに対し、実行命令フィールド 15 の MOVE 命令を単に実行するだけになる。

同様に、2 番目の命令セット 10 の Y フィールド 12 には、次の 3 番目の命令セット 10 の実行命令フィールド 15 の実行命令、MOVE および ADD の準備情報として、ソース側のレジスタを設定する命令が記述されている。このため、タイプフィールド 16 にはレジスタとイミディエイトが Y フィールド 12 に記述されていることが定義されている。

本例のプログラム 43 は、3 番目以降の命令セット 10 も上記と同様であり、3 番目の命令セット 10 のタイプフィールド 16 および Y フィールド 12 に、次の 4 番目の命令セット 10 の実行命令 15 の準備情報が記述されている。4 番目の命令セット 10 の実行命

令 1 5 は、比較処理 (CMP) と、条件分岐処理 (JCC) である。  
このため、3 番目の命令セット 1 0 では、そのタイプフィールド 1  
6 と Y フィールド 1 2 とにより、次の実行命令 1 5 で比較対象とな  
るレジスタ R 1 と # END のイミディエイトの値 (# F F F F F  
5 F F F H) と、分岐先 # L N E X T のアドレス (# 0 0 0 0 0 5 0  
0 H) が準備情報として記述されている。したがって、4 番目の命  
令セット 1 0 の実行命令 1 5 を実行するときは、比較回路として動  
作する演算処理ユニット 3 4 に入力値がセットされているので、そ  
のサイクルで比較結果を出す。また、ジャンプアドレスがフェッチ  
10 アドレスレジスタにセットされているので、実行命令 1 5 の条件分  
岐では、比較結果によって、そのサイクルで遷移先の命令セット 1  
0 をフェッチすることができる。

4 番目の命令セット 1 0 では、そのタイプフィールド 1 6 および  
Y フィールド 1 2 により、次の 5 番目の命令セット 1 0 の実行命令  
15 1 5 である比較処理 (CMP) と条件分岐処理 (JCC) の準備情  
報として、比較するレジスタの情報 (R 0 および R 1) と、分岐先  
# L O O P のアドレス (# 0 0 0 0 0 4 9 6 H) が記述されている。  
したがって、4 番目の命令セットと同様に、5 番目の命令セット 1  
0 を実行すると、すでに X フィールド 1 1 に記述された CMP と J  
20 CC を演算処理ユニット 3 4 で実行するインタフェースは整ってい  
るので、そのサイクルで比較および条件分岐処理が実行される。

その 5 番目の命令セット 1 0 の Y フィールド 1 2 には、次の 6 番  
目の命令セット 1 0 の実行命令である移行処理 (MOVE) および  
分岐処理 (JMP) の準備情報として、ソース側のレジスタ情報 (R  
25 1) と遷移先 # L O O P のアドレスが記述されている。したがっ  
て、6 番目の命令セット 1 0 を実行すると、そのサイクルでデータ  
をディスティネーションのレジスタ R 0 に格納し、遷移先の # L O  
O P のアドレスから命令をフェッチすることができる。

このように、本発明の命令セットによれば、実行命令と、その実

行命令を行うためのインタフェースなどを記述した準備命令とを分離することができ、さらに、準備命令を実行命令に先立ってフェッチされる命令セットに記述して処理することができる。したがって、各々の命令セットに記述された実行命令を行うときは、ALU34

5 のソース側にデータがリードされているので純粹に算術命令だけを行うようになる。このため、AC特性が良く、実行周波数特性が向上する。さらに、実行命令に対する前後の差はあるが、従来のパイプラインと同様に、命令フェッチ、レジスタデコード、処理実行などを段階的に行うことが可能であり、スループットも向上できる。

10 また、本例のプログラムは2命令を1命令セットに記述できるようになっているので、VLIWと同様にプログラムカウンタの近傍の複数の命令を並列実行することにより処理速度を向上することができる。

さらに、4番目の命令セットの実行命令フィールド15には条件分岐が記述されており、その分岐先のアドレスは、この命令セット

15 に先行する3番目の命令セットのYフィールド12に記述されている。したがって、4番目の命令セットを実行する際に、あるいはそれに先立ってフェッチレジスタに分岐先のアドレスをセットし、分岐条件が成立したときにペナルティなく分岐先の命令セットを

20 フェッチあるいは実行することができる。さらには、分岐先の命令をプリフェッチしておくことも可能であり、分岐先の実行命令を実行する準備を事前に整えておくことも可能となる。したがって、分岐先の命令であっても1クロックの無駄もなく実行することが可能であり、1クロック単位で処理を正確に定義することができる。

25 図9には、さらに、本発明の命令セット10のYフィールド12を用いてデータフローを定義し、そのデータフローにより上記と同様の処理を行う、本発明のプログラム44を示してある。このプログラム44に記述されたデータフロー指定命令25の内、DFLWIは、データフローの初期設定を行う命令であり、DFLWCはデ

ータフロー（データパス）を構成する演算処理ユニット34の接続情報（インタフェースの情報）および処理内容を規定する命令である。また、DFLWTはデータフローの終了条件を規定する命令であり、最後に、このようにして定義されたデータフローにデータを  
5 入力して処理を行うDFLWSが記述されている。これらのデータフロー指定命令25は、Yフィールド12に準備情報として記述され、第2の実行制御ユニット33でデコードされ、処理ユニット34でデータ処理を行うための構成（コンフィグレーション）がセットされる。

10 図9に示した本例のプログラム44を実行する際には、プログラムのデータフロー指定にしたがって第2の実行制御ユニット33が、第2の制御工程として処理ユニットの入力および／または出力インタフェースを、その処理ユニットの実行時期とは独立して設定し、さらに、処理ユニットの処理内容も規定する処理を行う。また、第  
15 2の実行制御ユニット33は、スケジューラ36としても機能し、第2の制御工程として各処理ユニットのインタフェースを維持するスケジュールを管理する。

このため、図10に示すように、スケジューラ36として機能する第2の実行制御ユニット33により、3つの演算処理ユニット3  
20 4のインタフェース（入出力）と、その処理内容が規定され、その状態あるいはコンフィグレーションが終了条件が成立するまで保持される。したがって、これらの演算処理ユニット34により構成されるデータフローあるいはデータパスにより、プログラムカウンタとは独立して次々と図6に示した処理と同じ処理が進行する。すな  
25 わち、データフロー指定を行うことにより、3つの演算処理ユニット34によって制御ユニット30の中に、その処理のための専用回路が事前に設けられた状態となり、プログラムカウンタの制御から外れて最大値を求める処理を実行することができる。そして、DP1. R1と#ENDが同じになることをDP1. SUBとしての機

能を果たすALU34で判断するとデータフローが終了する。

したがって、図9から判るように、データフローを定義することにより分岐命令を用いずに図6あるいは図7に記載されたプログラムを同じ処理を実行することができる。このため、汎用の制御ユニット30でありながら、専用回路を備えた制御ユニットと同様に特定の処理を非常に高速に効率良く行うことが可能となる。

本発明にかかる命令セットおよび制御ユニットにより、様々な処理を行うデータフローあるいは疑似データフローを制御ユニットに設けることができる。これらのデータフローはテンプレートとして他の処理あるいは他のプログラムにも適用できるものであり、ソフトウェアを用いてハードウェアを随時、特定のデータ処理に適した構成に変更でき、それを他のプログラムあるいは他のハードウェアにおいても実現できることを意味する。そして、このようなデータフローを複数設定することも可能であり、マルチコマンドストリームをソフトウェアを用いて制御ユニットの中に定義することができる。したがって、複数の処理を並列実行することが極めて簡単となり、その実行内容をプログラミングにより自由に制御することができる。

図11に、本例のXフィールド11およびYフィールド12を備えた命令セット10によりデータフローを定義することができる複数の処理ユニット（テンプレート）を備えたデータ処理装置の概略構成を、システムLSI50のイメージで示してある。このシステムLSI50は、データの処理動作を行うプロセッサ領域51と、そのプロセッサ領域51の処理を制御するプログラム18が格納されたコードRAM52と、その他の制御情報あるいは処理用のデータを記憶し、さらに、一次的なワーク領域ともなるデータRAM53とを備えている。プロセッサ領域51は、プログラムコードをフェッチするフェッチユニット（FU）55と、多目的な処理を行

う汎用的なデータ処理ユニット（多目的ALU、第1の制御ユニット）56と、データフロー方式でデータを処理することができるデータフロー処理ユニット（DFU、第2の制御ユニット）57とを備えている。

5       本例のLSI50は、1つの命令セット10に1組のXフィールド11およびYフィールド12を含んだプログラムコードをデコードして処理を実行できるようになっている。このため、FU55は、  
10       フェッチした命令セット10のXフィールド11の命令を格納できるフェッチレジスタ（FR（X））61xと、Yフィールド12の命令を格納できるフェッチレジスタ（FR（Y））61yとを備えている。また、FR（X）61xにラッチされた命令をデコードするXデコーダ62xと、FR（Y）61yにラッチされた命令をデコードするYデコーダ62yとを備えている。また、これらのデコーダ62xおよび62yのデコード結果により次の命令セットのアドレスが格納され、プログラムカウンタとして機能するレジスタ（PC）63を備えている。したがって、コードRAM52に格納されているプログラムの所定のアドレスから次の命令セットを随時フェッチすることができる。

20       本例のLSI50においては、Xデコーダ62xが上述した第1の実行制御ユニット32としての機能を果たす。したがって、Xデコーダ62xが、命令セット10のXフィールド11に記述された実行命令に基づき、本発明の第1の制御工程を実行する。また、Yデコーダ62yが第2の実行制御ユニット33としての機能を果たす。したがって、Yデコーダ62yが、命令セット10のYフィールド12に記述された準備情報に基づき、本発明の第2の制御工程を実行する。すなわち、本例のデータ処理装置の制御においては、  
25       フェッチユニット55において、本発明の命令セットをフェッチする工程が行われ、Xデコーダ62xにおいて、第1のフィールドの実行命令をデコードし、その実行命令の演算または他のデータ処理

が実行できるように予め設定された処理ユニットにより当該演算または他のデータ処理を進める第1の制御工程が行われ、Yデコーダ62yにおいて、第1の制御工程とは独立して、第2のフィールドの準備情報をデコードし処理ユニットの状態を演算または他のデータ処理が実行できるように設定する第2の制御工程が行われる。

多目的ALU56は、図5で説明した演算ユニット(ALU)34と、このALU34の入出力のデータを格納するレジスタ群35とを備えている。FU55でデコードされた命令がALU34の実行命令と準備情報であれば、Xデコーダ62xでデコードされた信号 $\phi_x$ と、Yデコーダ62yでデコードされた信号 $\phi_y$ は多目的ALU56に供給され、上記にて説明したようにALU34における処理が実行される。

DFU57は、様々な処理を行うデータフローあるいは疑似データフローを構成するための複数のテンプレート71が配置されたテンプレート領域72を備えている。それぞれのテンプレート71は、図9および図10に基づき説明したように、演算処理ユニット(ALU)などのような特定のデータパスあるいはデータフローとしての機能を備えている処理ユニット(処理回路)である。そして、Yフィールド12に準備情報として記述されたデータフロー指定命令25をYデコーダ62yがデコードし、その信号 $\phi_y$ により、DFU57の処理ユニットであるテンプレート71それぞれのインタフェースと処理内容を規定することができる。

したがって、これらのテンプレート71の接続および処理内容をYフィールド12に記述したデータフロー指定命令25によって変更することが可能である。このため、これらのテンプレート71の組み合わせにより、テンプレート領域72に特定のデータ処理に適したデータパスをプログラム18によりフレキシブルに構成することが可能となる。したがって、プロセッサ51の中に、特定の処理のための専用回路が設けられた状態となり、そこでの処理をプログ

ラムカウンタの制御から外れて実行することができる。すなわち、データフロー指定命令 25 によりテンプレート 7 1 の入出力と処理内容を変更することができるので、本例のプロセッサ 5 1 はソフトウェアを用いてハードウェアを随時、特定のデータ処理に適した構成に変更することができる。

図 1 2 (a) に示したように、本例のプロセッサ 5 1 の D F U 5 7 で入力データ  $\phi i n$  に処理を施して出力データ  $\phi o u t$  にする場合、たとえば、図 1 2 (b) に示すように、テンプレート 1-1、1-2 および 1-3 を直列に繋いであるデータ処理を行うようにテンプレート 7 1 のインタフェースをデータフロー指定命令 25 で設定することができる。同様に、テンプレート領域 7 2 の他のテンプレート 7 1 に対してもそれらのインタフェースをセットして複数のテンプレート 7 1 を適当に組み合わせてデータパスあるいはデータフローを構成することが可能であり、テンプレート領域 7 2 に入力データ  $\phi i n$  の処理に適した専用処理ユニットあるいは専用データパス 7 3 を複数個、プログラム 1 8 により随時構築することができる。

一方、入力データ  $\phi i n$  に対する処理が変わったときは、図 1 2 (c) に示すように、データフロー指定命令 25 によりテンプレート 7 1 の間の接続を変えることが可能である。すなわち、データフロー指定命令 25 を Y デコーダ 6 2 y がデコードし、該当するテンプレート 7 1 のインタフェースを変更することができる。このような Y デコーダ 6 2 y の制御（第 2 の制御工程）により、テンプレート 1-1、2-n および m-n を直列に接続して、他の異なる処理を実行するのに適した 1 つあるいは複数のデータパス 7 3 をテンプレート領域 7 2 に構築することが可能である。

また、テンプレート 7 1 を単独で、あるいは複数のテンプレート 7 1 を組み合わせて構成された処理ユニットは、並列して実行される他の処理あるいは他のプログラムに割り当てることも可能である。



複数のプロセッサ 5 1 が適当なバスで接続されていれば、他のプロセッサ 5 1 が主として行っているデータ処理のためにテンプレート 7 1 を組み合わせたトレイン（データパス） 7 3 を構成することも可能であり、テンプレート 7 1 というデータ処理資源を極めて有効に活用することができる。

さらに、ANDやORなどの単純な論理ゲートの実現をもカバーする目的のFPGAとは異なり、本発明に係るテンプレート 7 1 は、ALUなどとしての機能あるいは論理ゲートを基本的に備えた特定のデータパスを内部に実装する、より高いレベルのデータ処理ユニットである。そして、データフロー指定命令 2 5 により、テンプレート 7 1 のインタフェースを定義する、あるいは再定義することにより、それらの組み合わせを変えて特定の処理に適したさらに大きなデータパスを構成している。さらに、データフロー指定命令 2 5 によりテンプレート 7 1 で実行する処理内容を定義できるが、その際も、テンプレート 7 1 の内部のALUあるいは他の論理ゲートなどの接続を変更することで、テンプレート 7 1 の内部データパスの一部を選択する形で、テンプレート 7 1 で実行する処理内容を定義するようにしている。

したがって、本例のテンプレート 7 1 が複数配置されたDFU 5 7 のハードウェアを特定のデータ処理に適した構成に変更するときには、FPGAのようにチップ全体を、あるいは限定された論理ブロック単位でもマッピングしなおす必要はなく、テンプレート 7 1 あるいはテンプレート領域 7 2 に予め設けられたデータパスを切り替えたり、それらの一部を選択することにより、予め用意されたALUあるいは論理ゲートを用いて所望のデータパスを実現することができる。すなわち、テンプレート 7 1 の内部では論理ゲートのコネクションを必要な範囲で設定しなおし、テンプレート 7 1 の間でもそのコネクションを必要な範囲で設定し直すだけでよい。このため、極めて短時間に、クロック単位で、ハードウェアを特定のデ

ータ処理に適した構成に変更することができる。

さらに、論理ゲートが内蔵されていないFPGAでは、極めて汎用的である反面、特定のアプリケーションの機能を実現するロジック回路を形成するためには無駄となる配線も多く、冗長で信号経路も短くはならない。したがって、実行するアプリケーションに特化したASICに対して実装面積が大きくなり、また、AC特性も劣化する。これに対し、予め適当な論理ゲートを内蔵している本例のテンプレート71を採用したプロセッサ51では、FPGAのように膨大な無駄な領域が発生するのを防止でき、AC特性も改善することができる。したがって、テンプレート71をベースとした本例のデータ処理ユニット57は、ハードウェアをプログラムで変更可能なリコンフィグラブルな構成の処理装置であり、FPGAを採用した処理装置に対し、より高いレベルでソフトウェアのフレキシビリティとハードウェアの高速性とを備えたデータ処理装置を提供することができる。

そして、本例のテンプレート71は、適当な論理ゲートを予め内蔵しているので、特定のアプリケーションの処理を実現するために必要な論理ゲートを適当な実装密度で実現することができる。このため、テンプレート71を用いたデータ処理ユニットは経済的である。また、FPGAでデータ処理装置を構成した場合には、実装密度の低下をカバーするために、論理を再構成するプログラムのダウンロードを頻繁に行うことを検討する必要がある、そのための時間も処理速度が低下する原因となる。これに対し、本例のテンプレート71を用いたプロセッサ51では、実装密度が高いので、実装密度の低下をカバーする必然性は減少し、そのためにハードウェアを再構成する要求は少なくなる。そして、ハードウェアの再構成もクロック単位で制御することができる。これらの点でも、FPGAをベースとしたリコンフィグラブルな処理装置と異なり、ハードウェアをソフトウェアにより再構築できる処理装置であって、コンパクト

トで実行速度の速いデータ処理装置を提供することができる。

さらに、図 1 1 に示した D F U 5 7 は、テンプレート領域 7 2 に配置されたテンプレート 7 1 のインタフェースおよび処理内容（以降においてはコンフィグレーションデータ）を一括して定義あるいはセットすることができるコンフィグレーションレジスタ（C R E G）7 5 と、その C R E G 7 5 にセットする複数のコンフィグレーションデータ C i （i は適当な整数を示す、以下においても同様である）を記憶したコンフィグレーション R A M （C R A M）7 6 を備えている。そして、データフロー指定命令 2 5 として「D F S E T C i」といった命令が用意されており、Yデコーダ 6 2 y がこの命令をデコードすると、C R A M 7 6 に記憶されているコンフィグレーションデータ C i の中から所望のデータが C R E G 7 5 にロードされる。その結果、テンプレート領域 7 2 に配置された複数のテンプレート 7 1 のコンフィグレーションを一括して変更できる。あるいは、複数のテンプレート 7 1 からなる処理ブロック単位でそのコンフィグレーションを変更することができる。

また、D F L W I あるいは D F L W C といった上記のようなデータフロー指定命令 2 5 を Yデコーダ 6 2 y がデコードすることにより、個々のテンプレート 7 1 のコンフィグレーションを設定あるいは変更することも可能である。したがって、本例の D F U 5 7 では、多くの情報が必要となる複数のテンプレート 7 1 のコンフィグレーションを 1 命令で変更することが可能であり、命令効率がよく、さらに、再構成のために消費される時間が短縮されている。

さらに、本例の D F U 5 7 は、C R A M 7 6 にブロック単位でコンフィグレーションデータをダウンロードするコントローラ 7 7 を備えている。また、データフロー指定命令 2 5 として「D F L O A D B C i」が用意されており、Yデコーダ 6 2 y がこの命令をデコードすると、データ R A M 5 3 などに予め用意されている多数のコンフィグレーションデータ 7 8 の中から、進行中の処理あるいは

今後発生するであろう処理のためのコンフィグレーションデータ C  
i を予めコンフィグレーションメモリである C R A M 7 6 にダウ  
ロードしておくことができる。このような構成により C R A M 7 6  
に小容量の高速な連想メモリなどを採用することが可能となり、さ  
5 らに短時間でハードウェアをフレキシブルに変更することができる。

図 1 3 に、テンプレート 7 1 の一例を示してある。このテンプレ  
ート 7 1 は、D F U 5 7 に用意されたデータフロー R A M ( D F R  
A M ) 7 9 を介して他のテンプレート 7 1 とデータを交換すること  
ができる構成となっており、I / O インタフェース 8 1 を介して他  
10 のテンプレート 7 1 の処理結果が入力キャッシュ 8 2 a ~ 8 2 d に  
入力され、処理された結果が出力キャッシュ 8 3 a ~ 8 3 d に出力  
される。このテンプレート 7 1 は、これらの入力キャッシュ 8 2 a  
~ 8 2 d に各々ストアされたデータ A、B、C および D に対し以下  
の処理を実行し、演算結果は出力キャッシュ 8 3 b に、比較した結  
15 果は出力キャッシュ 8 3 c にストアすることができるデータパス 8  
8 を備えている。このテンプレート 7 1 の処理結果は、再び I / O  
インタフェース 8 1 および D F R A M 7 9 を介して他のテンプレ  
ートに出力される。

IF A == ?

20 THEN (C+B)==D

ELSE (C-B)==D . . . (A)

このテンプレート 7 1 は、独自のコンフィグレーションレジスタ  
8 4 を備えており、このレジスタ 8 4 に格納されるデータによって  
複数のセレクト 8 9 を制御し、制御部 8 5、加算器 8 6、比較器 8  
7 などの論理ゲートに入力する信号を選択することができる。した  
25 がって、テンプレート 7 1 は、コンフィグレーションレジスタ 8 4  
のデータを変更することにより、データパス 8 8 の一部を用いた処  
理も可能であり、たとえば、制御部 8 5 を用いずに、以下のような  
処理を実行させることも可能である。

$$(B+C)==D$$

$$(B-C)==D \quad \cdot \cdot \cdot (B)$$

また、同様にコンフィグレーションレジスタ 8 4 のデータを変えることにより、このテンプレート 7 1 は、データパス 8 8 の一部を用いて、制御部 8 5 による条件判定回路、加算器 8 6 を用いた加減演算回路、比較器 8 7 を用いた比較回路としても使用することができる。これらの論理ゲートはテンプレート 7 1 に予め作りこまれた専用回路で構成されているので、回路構成としても、処理時間としても無駄がない。そして、入力および出力データのコンフィグレーションは、コンフィグレーションレジスタ 8 4 によって制御されるインタフェース 8 1 により変更することが可能であり、所望のデータ処理を行うデータフローの全部あるいは一部を、本例のテンプレート 7 1 で処理することができる。

このテンプレート 7 1 は、さらに、独自のコンフィグレーションレジスタ 8 4 のデータを上述した C R E G 7 5 からのデータと、F U 5 5 の Y デコーダ (Y D E C) 6 2 y からのデータのいずれに基づいても書き換えることが可能であり、その選択は Y デコーダ 6 2 y からの信号により制御することができる。すなわち、上述したようなテンプレート 7 1 のコンフィグレーションは、データフロー指定命令 2 5 に基づき Y デコーダ 6 2 y あるいはこの Y デコーダ 6 2 y で実行される第 2 の制御工程によって行うことができる。さらに、D F S E T 命令などにより C R A M 7 6 に記憶されたコンフィグレーションデータ C i にしたがって、他のテンプレートと共にコンフィグレーションを変えてハードウェア構成を変更することも可能である。また、データフロー指定命令 2 5 によりコンフィグレーションレジスタ 8 4 のデータを設定できるので、テンプレート 7 1 の特定のデータパス 8 8 を部分的に選択して使用することも可能である。

このため、テンプレート 7 1 を個別でもグループあるいはブロッ

ク単位でもデータフロー指定命令 25 によってコンフィグレーションを変え、プロセッサ 51 のデータパスをフレキシブルに構成することができる。

テンプレート 71 の構成は本例に限定されるものではなく、他の  
5 データ処理を実現可能なように論理ゲートを組み合わせた、適当な  
種類と数のテンプレートを用意しておくことにより、それらの組み  
合わせを変えたり、処理内容の一部を変更することにより、多くの  
データ処理をテンプレート 71 を組み合わせたデータパスにより処  
理することができる。すなわち、本発明によれば、ある程度コンパ  
10 クトなデータパスを幾種類かのテンプレートとして用意しておき、  
そのデータパス間の組み合わせを指示して、データフロー型の処理  
に持ち込むことにより高性能化を図ることが可能である。そして、  
テンプレートでは対応できない処理は、プロセッサ 51 の多目的 A  
L U 56 の機能を用いて実行することが可能である。さらに、本例  
15 の多目的 A L U 56 は命令セット 10 の Y フィールド 12 に記述さ  
れた準備命令により分岐などにより発生するペナルティを最小限に  
止められるようになっている。このため、本例のプロセッサ 51 を  
搭載したシステム L S I 50 により、プログラムで処理を記述する  
のと同様に柔軟にハードウェアを変更し、高速処理あるいはリアル  
20 タイム処理が可能な高性能の L S I を提供することができる。また、  
アプリケーションの変更や仕様変更などに対して柔軟に対応でき、  
仕様変更などに伴い処理性能が低下することも防止できる。

システム L S I 50 を開発あるいは設計する時点で、システム L  
S I 50 を用いて実行するアプリケーションの概要が判明している  
25 場合には、そのアプリケーションの処理に適した構成のテンプレ  
ートを中心にテンプレート領域 72 を構成することが可能であり、よ  
り多くのデータ処理をデータフロー型の処理で実行し、処理性能を  
高めることが可能である。汎用的な L S I を提供する場合には、浮  
動小数点演算、乗除算、画像処理などの汎用のアプリケーションで

多く発生する処理に適したテンプレートを中心にテンプレート領域 7 2 を構成することが可能である。

このように、本発明にかかる命令セットおよびデータ処理装置により、様々な処理を行うデータフローあるいは疑似データフローを備えた L S I を提供することが可能であり、ソフトウェアを用いてデータフローを実行するハードウェアを随時、特定のデータ処理に適した構成に変更できる。また、上記に説明した、テンプレートの組み合わせによりデータフロー型の処理を実行するアーキテクチャ、すなわち、D F U 5 7 あるいはテンプレート領域 7 2 は、X フィールド 1 1 および Y フィールド 1 2 を備えた命令セット 1 0 とは独立して、制御ユニットあるいはプロセッサなどのデータ処理装置に組み込むことが可能である。そして、F P G A よりも高速処理が可能であり、ハードウェアの変更に係る時間も短く、A C 特性も良いデータ処理装置を提供することができる。

また、本例の D F U 5 7 あるいはテンプレート領域 7 2 を、従来型の汎用の組込プロセッサ、すなわち、ニーモニックなコードで動作するプロセッサと共に組み込んでシステム L S I を構成することも可能であり、テンプレート 7 1 で対応できない処理は、汎用のプロセッサで処理することができる。しかしながら、従来のプロセッサでは、分岐のペナルティや、演算処理のためのレジスタを準備するためにクロックを消費するなどの問題があることは上述した通りであり、本例の X - Y フィールドを備えた命令セット 1 0 をデコードして実行できるプロセッサ 5 1 のような形態が望ましい。

さらに、本例のプロセッサ 5 1 および命令セット 1 0 であれば、Y フィールド 1 2 を用い、他の処理と並列して、D F U 5 7 のコンフィグレーションをデータ処理を実行する前に設定あるいは変更することが可能であり、処理効率およびプログラム効率の面で優れている。従来のニーモニックな命令コードと、データフロー型の命令コードとを 1 つの命令セットに記述することによりプログラム効率

を高めることも可能である。しかしながら、本例の命令セット 10 の Y フィールド 12 の機能は、データフロー型の命令コードを記述するだけでないことは上述したとおりである。

また、本発明に係るプロセッサは、Y フィールド 12 により実行  
5 に先立って物理的なデータパスの構成を変えることができる。これ  
に対し、従来のプロセッサでは、複数のマルチプロセッサ間の接続  
方法が、共有メモリ等を通す方法しか存在せず、アイドル状態のプ  
ロセッサが存在しても、その内部のデータ処理ユニットを外部から  
10 利用する方法が無かった。本発明にかかるデータ処理装置において  
は、適当なデータフローを設定することにより、余っているハード  
ウェアを他の制御ユニットあるいはデータ処理装置により使用する  
といったことも可能となる。

さらに、副次的な効果として、命令実行シーケンスの効率化と内  
部データパスの独立性の確保と自由度（流用度）の向上により、本  
15 発明にかかる制御ユニットあるいはそれを用いたプロセッサにおい  
ては、実行するハードウェアに余裕さえあれば、全く性質の異なる  
コンテキストの命令シーケンスを同時に供給しても問題無く実行す  
ることが可能となる。

更に、現在、ハードウェアとソフトウェアの強調設計によるメ  
20 リットが盛んに指摘されるようになったが、本発明による命令セッ  
トおよび制御ユニットを採用することにより、ユーザ側の要求する  
アルゴリズムやデータ処理を許されるハードウェア・コストでどう  
効率良く経済的に実現可能かという事に対しする 1 つの回答を与え  
ることができる。例えば、ハードウェア・コストを最小に抑制しな  
25 がら、性能向上に貢献可能なデータパス（データフロー）を、過去  
のデータパスに関する構成結果情報である本発明にかかる命令セッ  
ト（旧 D A P / D N A）のデータ情報と、その後に追加されるハー  
ドウェア構成情報およびデータ処理を実行するシーケンス情報から  
新しいタイプの組み合わせ結果、すなわち、新しいデータフローを



定義するソフトウェアを導き、極めて無駄の少ない最適解を提供することが可能となる。

また、従来は、ハードウェア構成が要素化され難いために、その相互の組み合わせ自体の柔軟性が無く、基本的には、性能を上げるために1つ新規のデータパスを追加するというようなやり方が主流であった。そして、性能向上のための情報蓄積の点でも、実際にそれを実現する上で必要となるハードウェア情報の追加という観点でも、数値化し難くデータベース化することは困難であった。これに対し、本発明によれば、ある程度コンパクトなデータパスをいくつかテンプレート的に用意しておき、そのデータパス間の組み合わせを指示して、データフロー型の処理に持ち込むことにより高性能化を図ることが可能である。そして、極めて細かい単位でのハードウェアとソフトウェアとの連携の見積もりが容易となる。また、ハードウェアとソフトウェアのトレードオフ情報を蓄積することも可能で、データパス単位でその組み合わせの可能性が、処理性能に対する貢献度と密接に結びつくことになる。したがって、ハードウェアとソフトウェアの緊密な実行性能データや処理要求に応じた性能コストの正確な見積もりを蓄積することが可能となる。もちろん、これらのデータパスは主要な処理あるいは汎用的な処理の実行を停止させないで実現することも可能となるため、性能要求に対して、何をどれだけどのように追加すれば、どのような結果が期待出来るということを、純粹に過去に蓄積された本発明にかかる命令セットおよびハードウェアのデータから予測する事が可能とする。

これは、現在行われている設計コストや仕様策定コストの著しい低減に貢献するだけで無く、次の新しい設計に対して、新規に追加すべきハードウェアとソフトウェアのトレードオフを必要最小限で完了させる事に貢献する。また、処理形態に応じて、内部のデータパスを外部へ貸し出しする事も容易にする為、ハードウェアのリソースシェアリング化が可能となり、複数の本発明にかかるモジュール

ル（DAP/DNAモジュール）の間で並列処理化を極め、コンパクトなハードウェアで実現する事が可能となる。

なお、上記に示したデータ処理装置および命令セットなどは、本発明の一例に過ぎず、たとえば、データ処理装置においては、コードRAMあるいはデータRAMなどを外部のRAMあるいはROMとしたり、これらに加えて外部のDRAMあるいはSRAMなどとのインタフェースを設けることも可能である。さらに、外部の他のデバイスと接続するための入出力インタフェースなど、システムLSIなどのデータ処理装置として公知の機能を備えたデータ処理装置も本発明に含まれる。したがって、本発明は以下の請求の範囲の記載により理解および把握され、それらの請求の範囲に含まれる変形例は全て本発明の範囲に含まれる。

また、本発明の命令セットおよびデータ処理装置により提供される新しいプログラミング環境においては、上述した以外にも特殊な命令を設けることが可能である。例えば、現在のプログラムとは別に、1つ以上のオブジェクト（プログラム）を同時に起動し、並列処理起動を命令レベルでサポートするXFORK、オブジェクト（プログラム）間の同期を指定するXSYNK、並列処理間のパイプライン結合を命令するXPIPE、現在のオブジェクトを終了し、次のオブジェクトを起動するXSWITCHなどが考えられている。

以上に説明したように、本発明にかかる命令セットおよびそれを用いたプログラミングおよびそれを実行可能なデータ処理装置の技術は、従来の命令セットの構成方法そのものを大幅に変更するものであり、これにより、従来技術では対応の難しかった上述したような問題を上手く解決し、大きな性能向上を図ることができる。

すなわち、本発明にかかる命令セットは、命令セットの構成方法を従来の命令セットの構成方法とは全く異なる視点から見直すことにより、従来技術では解決の極めて困難と思われる多くの問題を、

極めて効率良く解決している。実際、従来技術においては、その命令セットの構成法とハードウェアによる命令供給（入手）方法が、極めて画一的で伝統的な先入観により実現されていたため、本質的な意味での解決を遠ざけており、その問題点を全て膨大で複雑なハードウェア構成により解決しようとすることで社会へ貢献すべきテクノロジーとその上に構築される各種の情報処理製品の開発コストを膨大に引き上げる原因となっていた。本発明は、これを本来あるべきアプリケーション要求を優先した命令セットを実現することにより、単に製品性能の効率化に止まらず、その高い開発効率と製品の品質保証を得やすい手段を提供することができる。

また、本発明により、性能向上に貢献可能なデータパス（データフロー）をテンプレートという資産と、それを使用する命令セットという資産で蓄積できる。さらに、その後に追加されるハードウェア構成情報およびデータ処理を実行するシーケンス情報に基づき随時更新し最適解を求めるようにすることができる。したがって、従来存在したアプリケーション間の資産の共有化とハードウェア資産の共有化、及び高性能化に対する適切なハードウェア投資がより健全な方向へ向かい、ネットワーク化社会を構築する上でのテクノロジー・インフラとしても大きく貢献可能となることが期待できる。

#### 産業上の利用可能性

本発明のデータ処理装置は、様々なデータ処理を実行可能なプロセッサあるいはLSIなどとして提供することが可能であり、電子素子の集積回路のみならず、光素子、さらには電子素子および光素子を集積した光集積回路装置にも適用することができる。特に、本発明の命令セットを備えた制御プログラムおよびデータ処理装置においては、データ処理を柔軟に、そして高速に実行できるので、ネットワーク処理や、画像処理などの高速性およびリアルタイム性能を要求されるデータ処理装置に好適なものである。

## 請 求 の 範 囲

1. データ処理装置を構成する少なくとも1つの処理ユニットで  
実行する演算または他のデータ処理の内容を指示する実行命令を記  
述可能な第1のフィールドと、前記実行命令で実行する演算または  
5 他のデータ処理が実行可能な状態に前記処理ユニットを設定する準備  
情報を記述可能な第2のフィールドとを備えた命令セットを有する  
制御プログラム製品であって、

前記第1のフィールドに記述された前記実行命令の内容に対し独立  
した演算または他のデータ処理の前記準備情報が前記第2のフ  
10 イールドに記述された前記命令セットを有する制御プログラム製品。

2. 前記第2のフィールドに、後続の前記命令セットの前記第1  
のフィールドに記述された前記実行命令を実行するための前記準備  
15 情報が記述された前記命令セットを有する、請求項1の制御プログラム  
製品。

3. 前記準備情報により、前記処理ユニットの入力および／または  
出力インタフェースを、その処理ユニットの実行時期とは独立し  
て指示可能する、請求項1の制御プログラム製品。  
20

4. 前記準備情報により、前記処理ユニットの処理内容を指示可  
能である、請求項1の制御プログラム製品。

5. 前記データ処理装置は、複数の前記処理ユニットを備えてお  
り、前記準備情報により、前記処理ユニットによるデータパスの組  
み合わせを指示可能である、請求項1の制御プログラム製品。  
25

6. 前記処理ユニットは、特定の内部データパスを備えており、

前記準備情報により、前記内部データパスの一部を選択可能である、請求項 1 の制御プログラム製品。

5 7. 前記準備情報により、複数の前記処理ユニットにより構成される処理ブロックの入力および／または出力インタフェースを指示可能である、請求項 1 の制御プログラム製品。

10 8. 前記データ処理装置は、前記処理ブロックの入力および／または出力インタフェースを規定する複数のコンフィグレーションデータを格納したメモリを有し、

前記準備情報により、前記メモリに格納された前記複数のコンフィグレーションデータの 1 つを選択し、前記処理ブロックの入力および／または出力インタフェースを変更可能である、請求項 7 のプログラム製品。

15

9. 前記データ処理装置は、前記処理ユニットとして算術論理演算ユニットを備えた第 1 の制御ユニットと、前記処理ユニットとして特定の内部データパスを具備する複数のデータフロー処理ユニットを備えた第 2 の制御ユニットとを有し、

20 前記第 1 のフィールドに、前記算術論理演算ユニットを操作する前記実行命令が記述され、前記第 2 のフィールドに、前記算術論理演算ユニットおよび／または前記データフロー処理ユニットのインタフェースを規定する前記準備情報が記述された前記命令セットを有する、請求項 1 の制御プログラム製品。

25

10. 前記準備情報により、前記データフロー処理ユニットによるデータパスの組み合わせを指示可能である、請求項 9 の制御プログラム製品。

1 1. 前記準備情報により、前記内部データパスの一部を選択可能である、請求項 9 の制御プログラム製品。

5 1 2. 前記第 2 のフィールドに、レジスタまたはバッファとメモリの間の入出力を指示する命令が記述された、請求項 1 の制御プログラム製品。

10 1 3. 前記第 1 および／または第 2 のフィールドは、複数の前記実行命令または前記準備情報を記述可能なフィールドを備えている前記命令セットを有する、請求項 1 の制御プログラム製品。

15 1 4. データ処理装置を構成する少なくとも 1 つの処理ユニットで実行する演算または他のデータ処理の内容を指示する実行命令を記述可能な第 1 のフィールドと、前記実行命令で実行する演算または他のデータ処理が実行可能な状態に前記処理ユニットを設定する準備情報を記述可能な第 2 のフィールドとを備えた命令セットを有する制御プログラムであって、

20 前記第 1 のフィールドに記述された前記実行命令の内容に対し独立した演算または他のデータ処理の前記準備情報が前記第 2 のフィールドに記述された前記命令セットを有する制御プログラムが記録されている記録媒体。

25 1 5. データ処理装置を構成する少なくとも 1 つの処理ユニットで実行する演算または他のデータ処理の内容を指示する実行命令を記述可能な第 1 のフィールドと、前記実行命令で実行する演算または他のデータ処理が実行可能な状態に前記処理ユニットを設定する準備情報を記述可能な第 2 のフィールドとを備えた命令セットを有する制御プログラムであって、

前記第 1 のフィールドに記述された前記実行命令の内容に対し独

立した演算または他のデータ処理の前記準備情報が前記第2のフィールドに記述された前記命令セットを有する制御プログラムが埋め込まれている伝送媒体。

- 5      16.      演算または他のデータ処理を実行する少なくとも1つの処理ユニットと、

前記処理ユニットで実行する演算または他のデータ処理の内容を指示する実行命令を記述可能な第1のフィールド、および前記実行命令で実行する演算または他のデータ処理が実行可能な状態に前記  
10      処理ユニットを設定する準備情報を記述可能な第2のフィールドとを具備する命令セットをフェッチ可能なユニットと、

前記第1のフィールドの前記実行命令をデコードし、その実行命令の演算または他のデータ処理が実行できるように予め設定された前記処理ユニットにより当該演算または他のデータ処理を進める第  
15      1の実行制御ユニットと、

前記第2のフィールドの前記準備情報をデコードし、前記第1の実行制御ユニットの実行内容とは独立して前記処理ユニットの状態を演算または他のデータ処理が実行できるように設定する第2の実行制御ユニットとを有するデータ処理装置。

20

17.      前記第1または第2の実行制御ユニットは、前記第1または第2のフィールドに記述された複数の独立した前記実行命令または前記準備情報を独立して処理可能な複数の実行制御部を備えている、請求項16のデータ処理装置。

25

18.      前記第2の実行制御ユニットは、前記処理ユニットの入力および／または出力インタフェースを、その処理ユニットの実行時期とは独立して設定する、請求項16のデータ処理装置。

19. 前記第2の実行制御ユニットは、前記処理ユニットの処理内容を規定する、請求項16のデータ処理装置。

20. 複数の前記処理ユニットを有し、前記第2の実行制御ユニットは、これらの処理ユニットによるデータパスの組み合わせを制御する、請求項16のデータ処理装置。

21. 前記処理ユニットは、特定の内部データパスを備えている、請求項16のデータ処理装置。

22. 前記処理ユニットは、少なくとも1つの論理ゲートと、この論理ゲートと入出力インタフェースを接続する内部データパスとを備えている、請求項16のデータ処理装置。

23. 前記第2の実行制御ユニットは、前記準備情報により、前記処理ユニットの前記内部データパスの一部を選択可能である、請求項21のデータ処理装置。

24. 前記第2の実行制御ユニットは、前記準備情報により、複数の前記処理ユニットにより構成される処理ブロックの入力および／または出力インタフェースを変更可能である、請求項16のデータ処理装置。

25. 前記処理ブロックの入力および／または出力インタフェースを規定する複数のコンフィグレーションデータを格納したメモリを有し、

前記第2の実行制御ユニットは、前記準備情報により前記メモリに格納された前記複数のコンフィグレーションデータの1つを選択し、前記処理ブロックの入力および／または出力インタフェースを



変更可能である、請求項 24 のデータ処理装置。

26. 前記第 2 の実行制御ユニットは、前記処理ユニットのインタフェースを管理するスケジューラとしての機能を備えている、請求項 16 のデータ処理装置。

27. 前記処理ユニットとして算術論理演算ユニットを備えた第 1 の制御ユニットと、前記処理ユニットとして特定のデータパスを具備する複数のデータフロー処理ユニットを備えた第 2 の制御ユニットとをさらに有し、

前記第 1 の実行制御ユニットは、前記算術論理演算ユニットを操作し、

前記第 2 の実行制御ユニットは、前記算術論理演算ユニットおよび／または前記データフロー処理ユニットのインタフェースを設定する、請求項 16 のデータ処理装置。

28. 前記第 2 の実行制御ユニットは、前記データフロー処理ユニットによるデータパスの組み合わせを制御する、請求項 27 のデータ処理装置。

29. 前記データフロー処理ユニットは、特定の内部データパスを備えており、前記第 2 の実行制御ユニットは、前記準備情報により、前記データフロー処理ユニットの前記内部データパスの一部を選択可能である、請求項 27 のデータ処理装置。

30. 前記第 2 の実行制御ユニットは、レジスタまたはバッファとメモリの間の入出力を制御する機能を備えている、請求項 16 のデータ処理装置。

3 1. 演算または他のデータ処理を実行する少なくとも1つの処理ユニットを有するデータ処理装置の制御方法であって、

前記処理ユニットで実行する演算または他のデータ処理の内容を指示する実行命令を記述可能な第1のフィールド、および前記実行命令で実行する演算または他のデータ処理が実行可能な状態に前記処理ユニットを設定する準備情報を記述可能な第2のフィールドとを具備する命令セットをフェッチする工程と、

前記第1のフィールドの前記実行命令をデコードし、その実行命令の演算または他のデータ処理が実行できるように予め設定された前記処理ユニットにより当該演算または他のデータ処理を進める第1の制御工程と、

この第1の制御工程とは独立して、前記第2のフィールドの前記準備情報をデコードし前記処理ユニットの状態を演算または他のデータ処理が実行できるように設定する第2の制御工程とを有するデータ処理装置の制御方法。

3 2. 前記第2の制御工程では、前記処理ユニットの入力および／または出力インタフェースを、その処理ユニットの実行時期とは独立して設定する、請求項3 1のデータ処理装置の制御方法。

3 3. 前記第2の制御工程では、前記処理ユニットの処理内容を規定する、請求項3 1のデータ処理装置の制御方法。

3 4. 前記データ処理装置は、複数の前記処理ユニットを有し、前記第2の制御工程では、これらの処理ユニットによるデータパスの組み合わせを制御する、請求項3 1のデータ処理装置の制御方法。

3 5. 前記処理ユニットは、特定の内部データパスを備えており、前記第2の制御工程では、前記処理ユニットの前記内部データパ

スの一部を選択可能である、請求項 3 1 のデータ処理装置の制御方法。

5 3 6. 前記第 2 の制御工程では、複数の前記処理ユニットにより構成される処理ブロックの入力および／または出力インタフェースを変更可能である、請求項 3 1 のデータ処理装置の制御方法。

10 3 7. 前記データ処理装置は、前記処理ブロックの入力および／または出力インタフェースを規定する複数のコンフィグレーションデータを格納したメモリを有し、

前記第 2 の制御工程では、前記メモリに格納された前記複数のコンフィグレーションデータの 1 つを選択し、前記処理ブロックの入力および／または出力インタフェースを変更可能である、請求項 3 1 のデータ処理装置の制御方法。

15

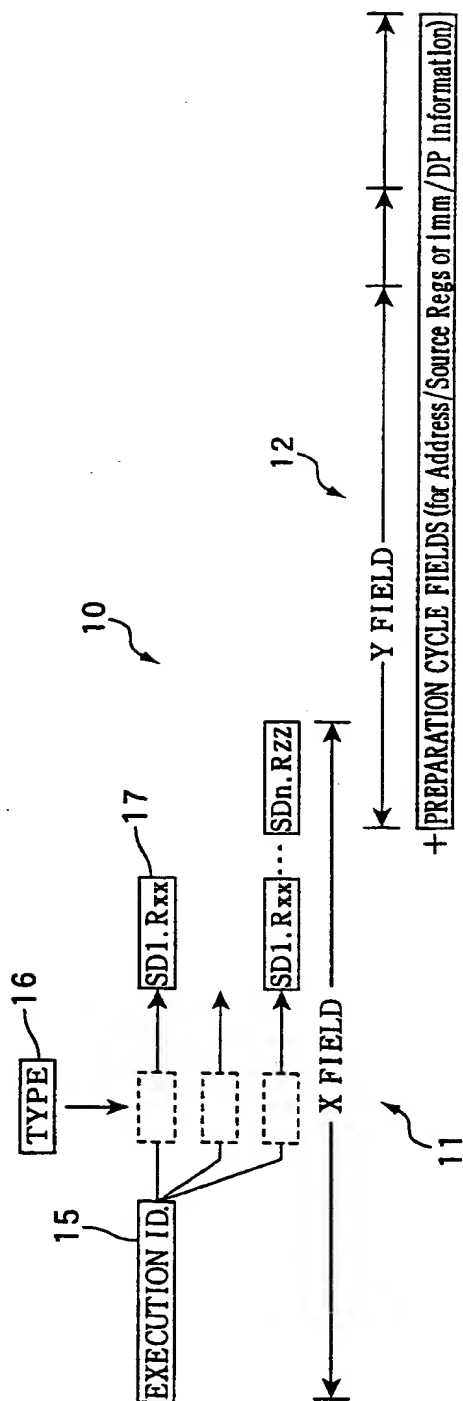
3 8. 前記第 2 の制御工程では、前記処理ユニットのインタフェースを保持するスケジュールを管理する、請求項 3 1 のデータ処理装置の制御方法。

20 3 9. 前記第 2 の制御工程では、レジスタまたはバッファとメモリの間の入出力を制御する、請求項 3 1 のデータ処理装置の制御方法。

**THIS PAGE BLANK (USPTO)**

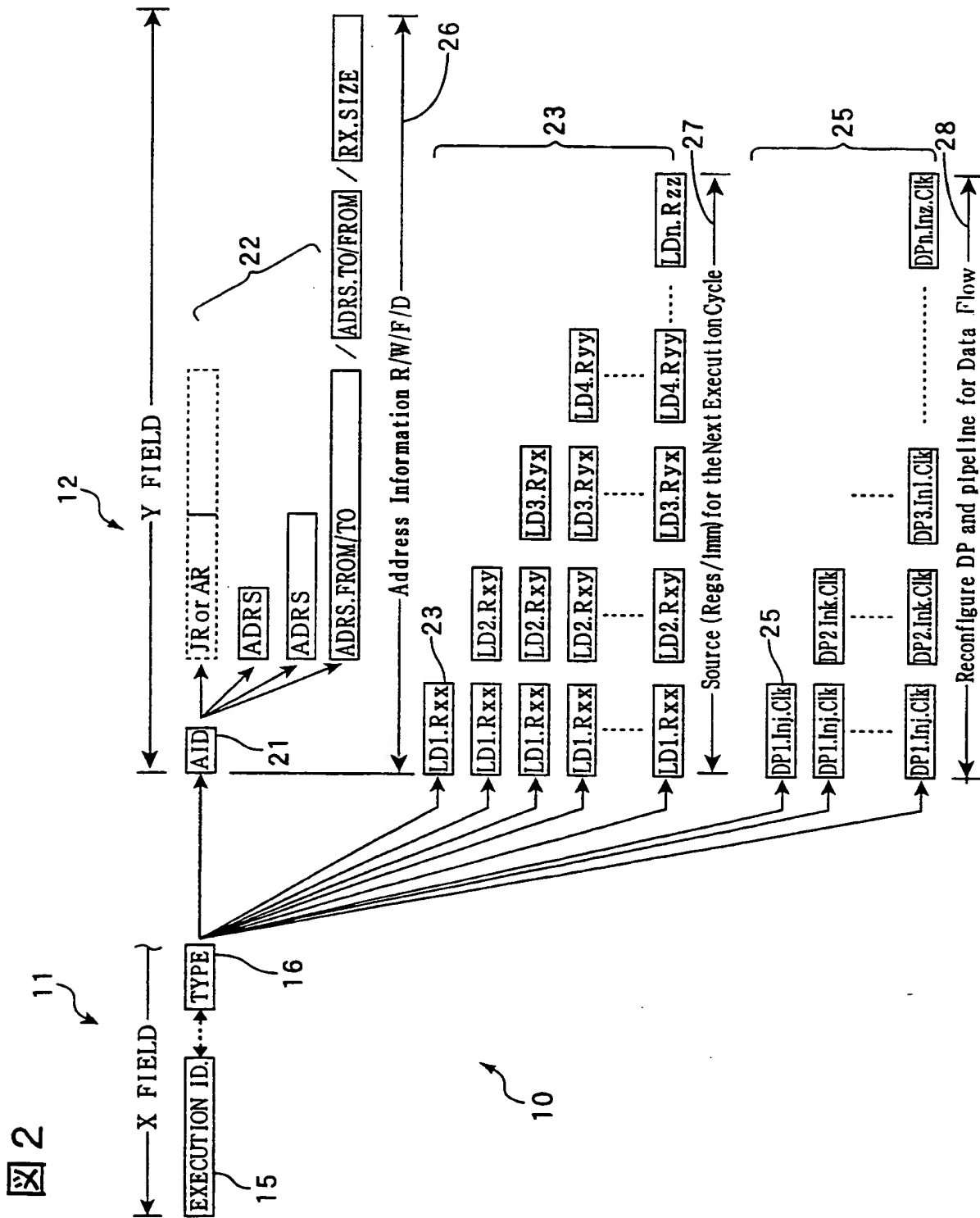
1 / 1 2

図 1



**THIS PAGE BLANK (USPTO)**

2 / 1 2



**THIS PAGE BLANK (USPTO)**



3 / 1 2

図 3

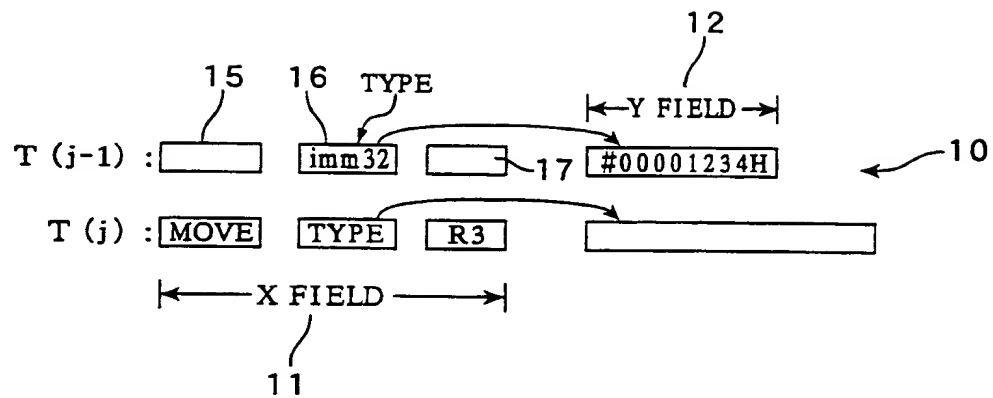
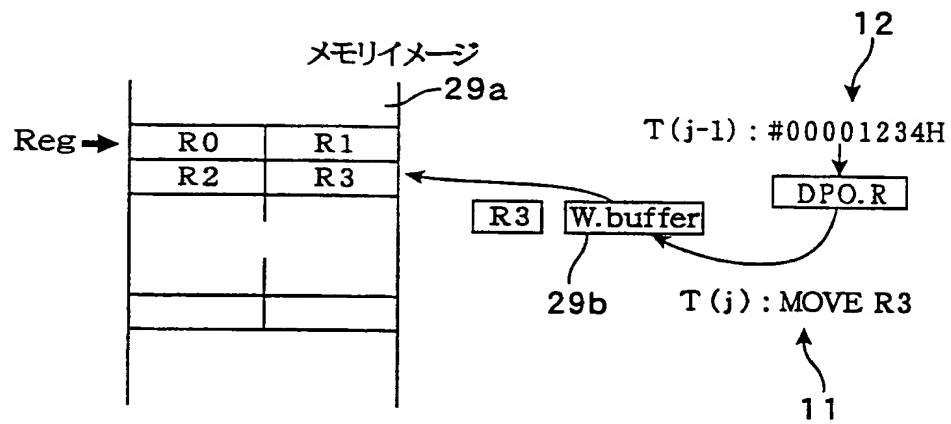


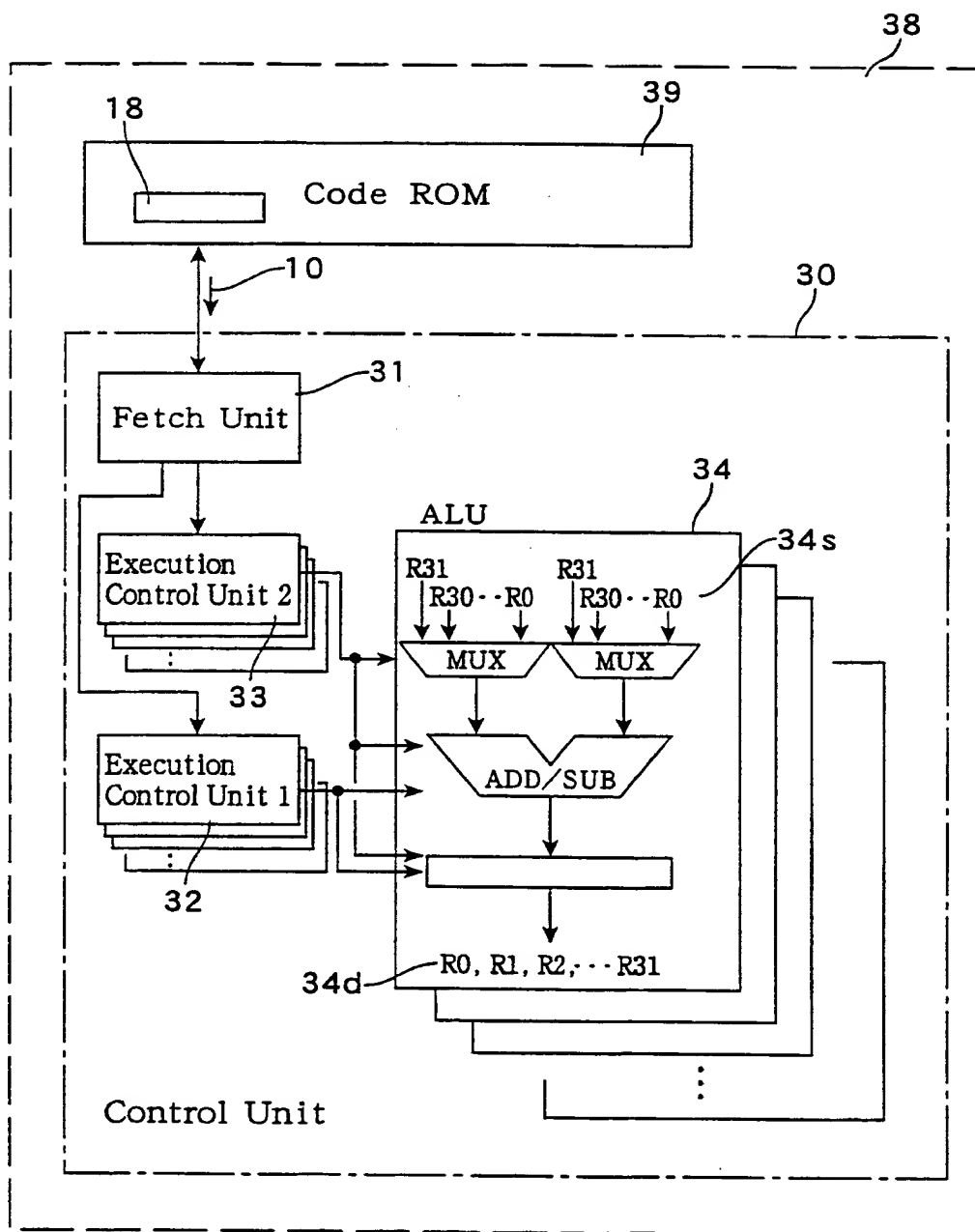
図 4



**THIS PAGE BLANK (USPTO)**

4 / 1 2

図 5



**THIS PAGE BLANK (USPTO)**

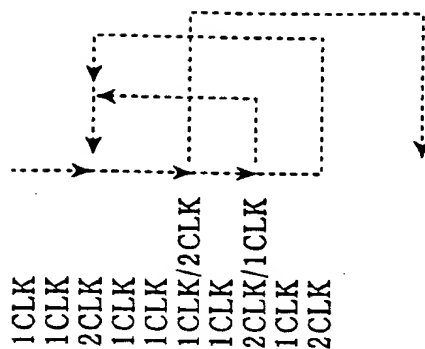
5 / 1 2

図 6

41

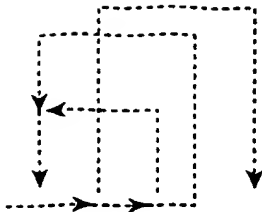
```

    MOVE R0, #00000000H:
    MOVE R3, #START : start address of the search table.
    MOVE R1, [R3] : store [R3] into R1
    ADD R3, #0001 : increment R3
    CMP R1, #END : finished?
    JEQ LNEXT : try a next program
    CMP R0, R1 : R0 > R1?
    JHI LOOP : compare the next one in this table.
    MOVE R0, R1 : Loop until the end data of the table.
    JMP LNEXT :
    LNEXT :
  
```



**THIS PAGE BLANK (USPTO)**

6 / 1 2



1CLK  
1CLK  
1CLK  
1CLK  
1CLK

7

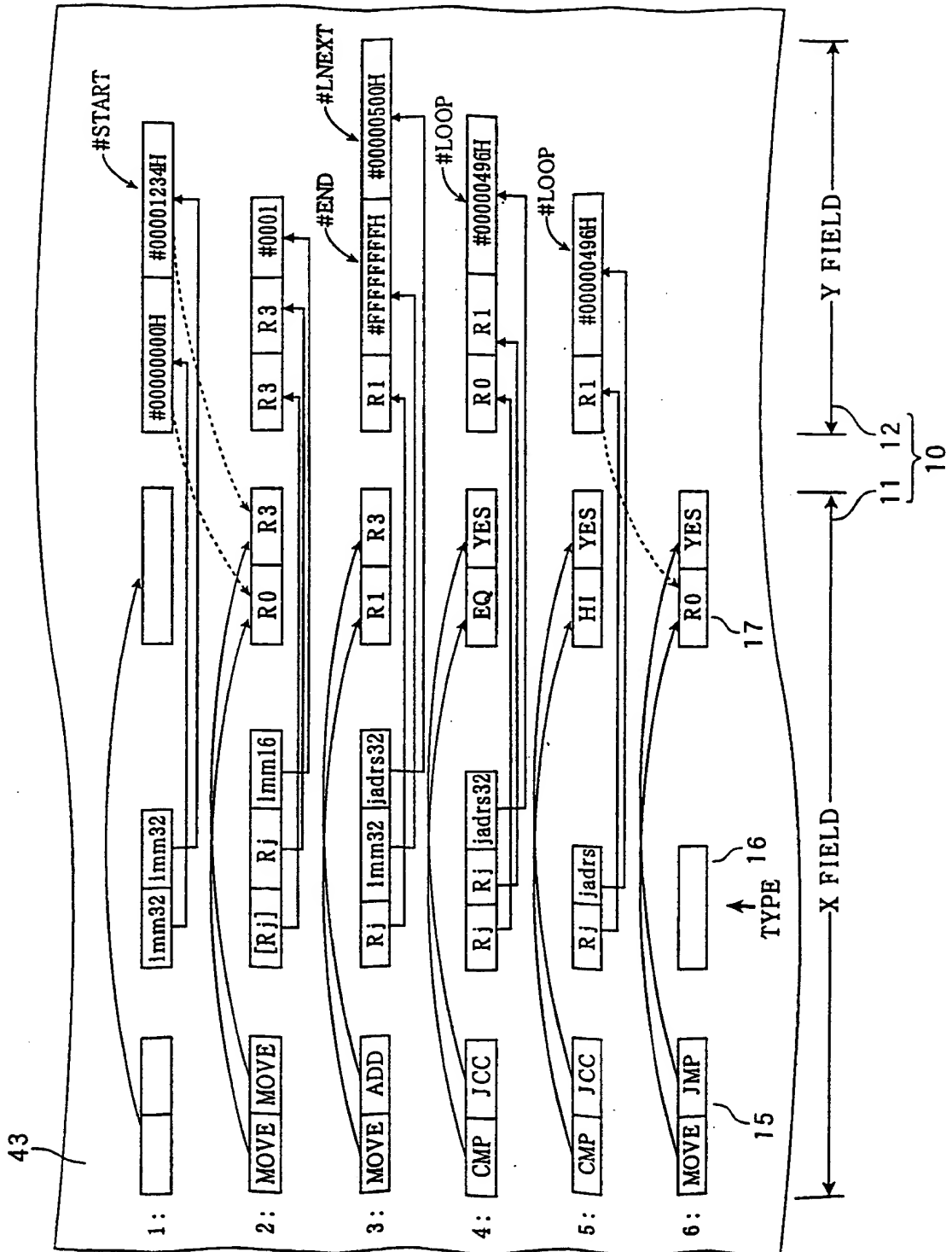
42

```
LOOP:  MOVE R0, #00000000H"  MOVE R3, #START.  
        MOVE R1, [R3]"      ADD  R3, R3, #0001.  
        CMP  EQ, R1, #END"   JYES LEXIT.  
        CMP  HI, R0, R1"     JYES LOOP.  
        MOVE R0, R1"        JMP  LOOP.  
  
LNEXT:
```

**THIS PAGE BLANK (USPTO)**



図 8



**THIS PAGE BLANK (USPTO)**

8 / 1 2

図 9

44

25

1CLK  
1CLK  
1CLK  
1CLK  
1CLK

```

[ DFLWI = MOVE DPL.R0, #00000000H"  MOVE DPL.R3, #START
  DFLWC = MOVE DPL.R1, [R3]"      ADD  DPL.R3, DPL.R3, #0001
  DFLWT = STOP.EQ  DPL.R1, #END.
  DFLWC = LDMAX, DPL.R0, DPL.R0, DPL.R1
  DFLWS = START

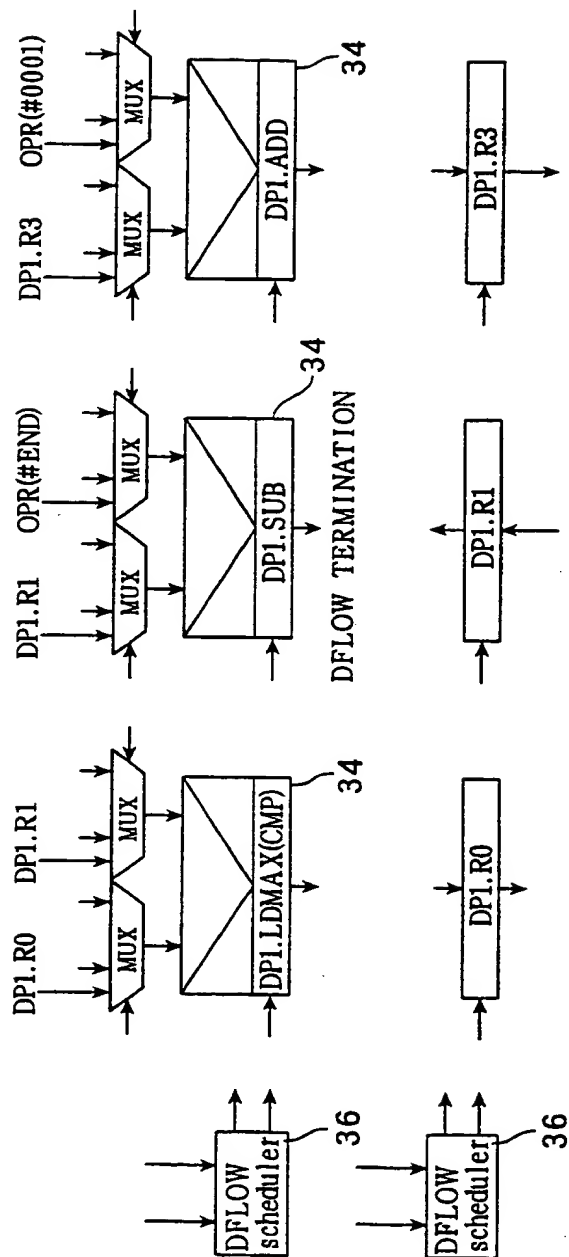
  MOVE R0, #012345678"  MOVE R1, #NEXT_JOB
  ADD  R5, R2, R3"      CALL FUNC_ABCD
  CMP  R0, R1
  :
  WAIT DFLW, DPL        ;wait for the termination of above data processing.
  MOVE R0, DPL.R0       ;get themax data in the search table.

```

----->

**THIS PAGE BLANK (USPTO)**

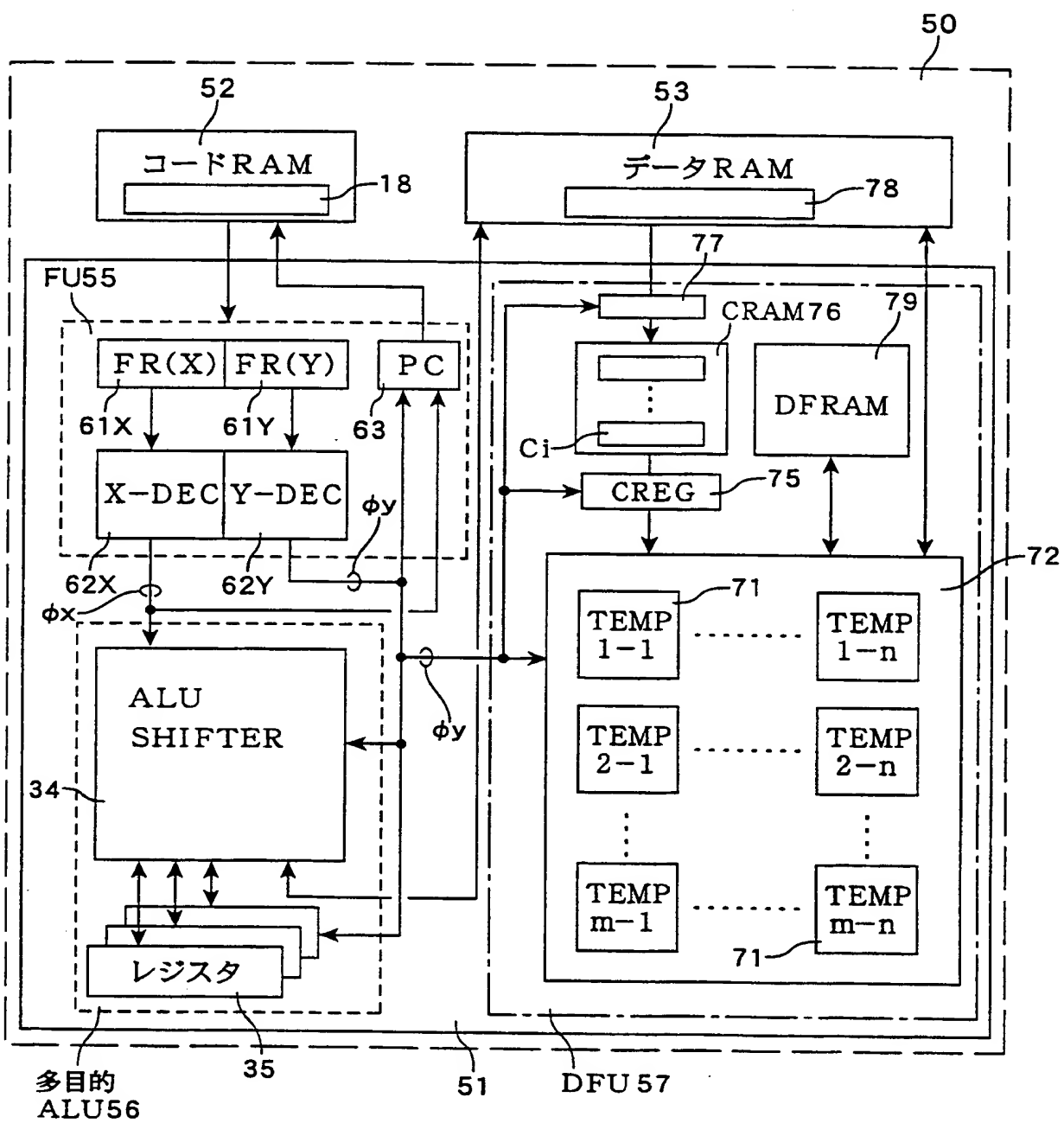
図 10



**THIS PAGE BLANK (USPTO)**

1 0 / 1 2

図 1 1

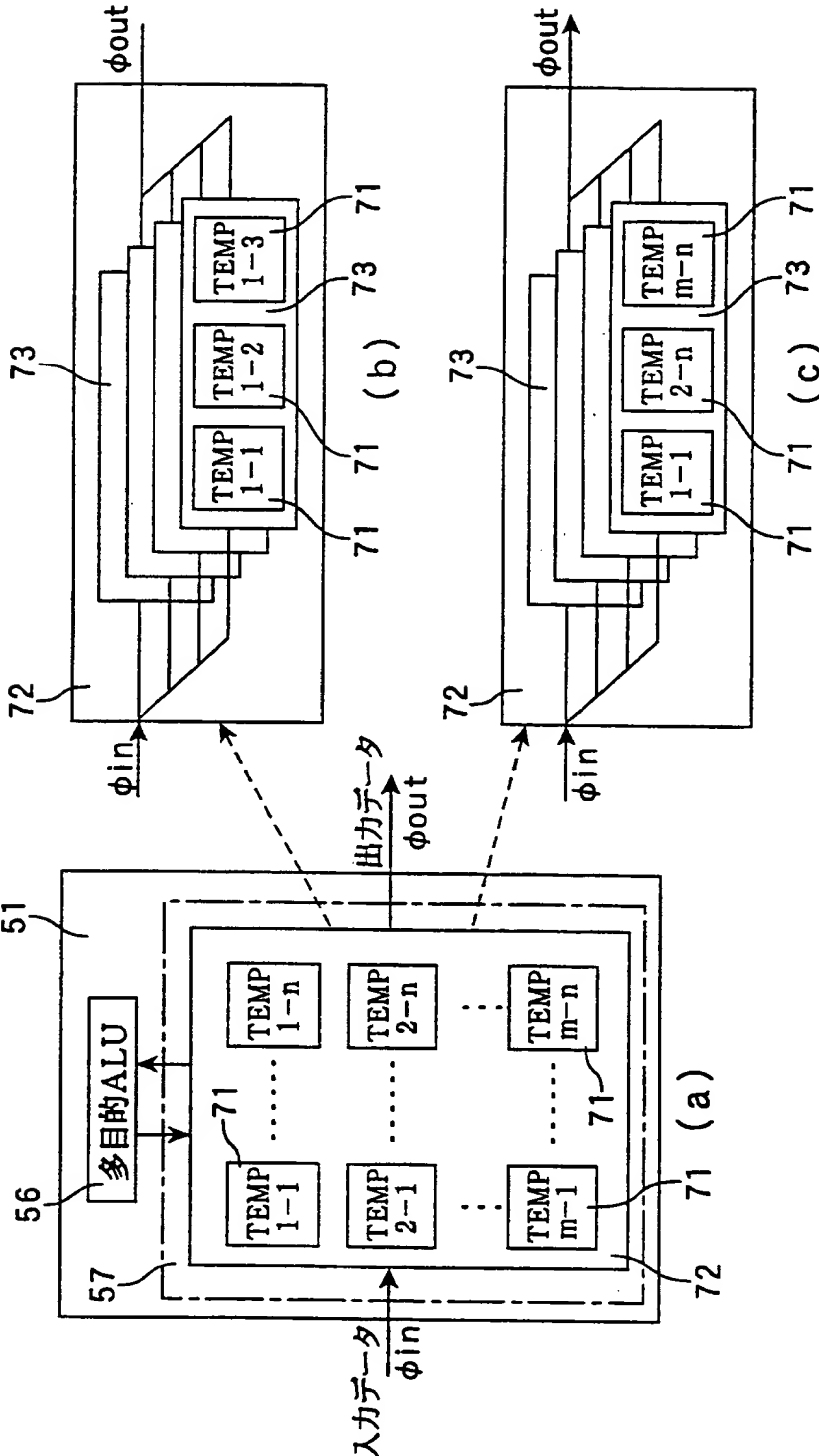


**THIS PAGE BLANK (USPTO)**



1 1 / 1 2

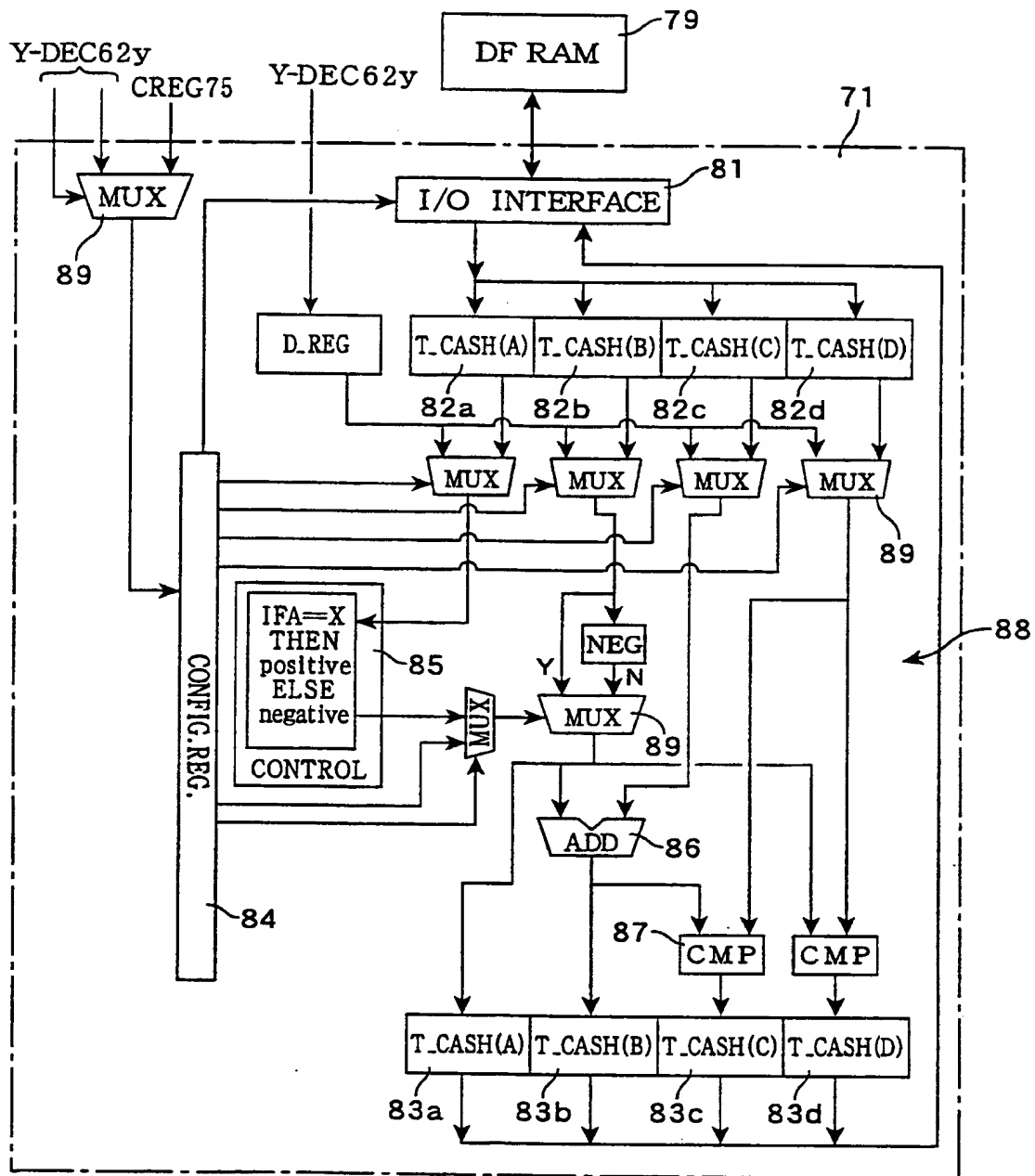
図 12



**THIS PAGE BLANK (USPTO)**

1 2 / 1 2

図 1 3



**THIS PAGE BLANK (USPTO)**

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP00/05848

A. CLASSIFICATION OF SUBJECT MATTER  
Int.Cl<sup>7</sup> G06F 9/30, G06F 9/38

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)  
Int.Cl<sup>7</sup> G06F 9/30 - 9/42

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched  
Jitsuyo Shinan Koho 1922-1996 Toroku Jitsuyo Shinan Koho 1994-2000  
Kokai Jitsuyo Shinan Koho 1971-2000 Jitsuyo Shinan Toroku Koho 1996-2000

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)  
CSDB (JAPANESE PATENT OFFICE),  
[FOLLOWING/CW\* [COMMAND/CW+PRECEDING/CW]]  
PATENT FILE (PATOLIS), [AP=IP FLEX INC.]

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X A	JP, 7-253882, A (Hitachi, Ltd.), 03 October, 1995 (03.10.95), Full text; Figs. 1 to 10 Full text; Figs. 1 to 10 (Family: none)	14, 16, 19, 31, 33 17, 18, 20-30, 32, 34-39
X A	JP, 1-156824, A (Hitachi, Ltd. et al.), 20 June, 1989 (20.06.89), Full text; Figs. 1 to 11 Full text; Figs. 1 to 11 (Family: none)	14, 16, 19, 31, 33 17, 18, 20-30, 32, 34-39
A	JP, 61-294548, A (Toshiba Corporation), 25 December, 1986 (25.12.86), Full text; Figs. 1 to 6 (Family: none)	14, 16-39
A	JP, 2-183332, A (Fujitsu Limited), 17 July, 1990 (17.07.90), Full text; Figs. 1 to 7 (Family: none)	14, 16-39

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

\* Special categories of cited documents:  
"A" document defining the general state of the art which is not considered to be of particular relevance  
"E" earlier document but published on or after the international filing date  
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)  
"O" document referring to an oral disclosure, use, exhibition or other means  
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention  
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone  
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art  
"&" document member of the same patent family

Date of the actual completion of the international search  
24 October, 2000 (24.10.00)

Date of mailing of the international search report  
31 October, 2000 (31.10.00)

Name and mailing address of the ISA/  
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP00/05848

## Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:  
because they relate to subject matter not required to be searched by this Authority, namely:
  
2. ☒ Claims Nos.: 1-13,15  
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
  1. The inventions of claims 1-13 relate to a control program product. However "product" that is a term the technical range of which is unclear is used, and therefore the inventions of claims 1-13 cannot be clearly understood.
  
3. ☐ Claims Nos.:  
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

## Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
  
2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
  
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
  
4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

☐  
☐

- The additional search fees were accompanied by the applicant's protest.  
No protest accompanied the payment of additional search fees.

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP00/05848

Continuation of Box No.I-2 of Continuation of first sheet (1)

2. The invention of claim 15 relates to a transmission medium where a control program is embedded, but the relation between the transmission medium and the program is not defined. Since the transmission medium, being a thing, cannot be specified by a transmitted program. Therefore the invention of claim 15 is not clearly understood.

**THIS PAGE BLANK (USPTO)**



A. 発明の属する分野の分類 (国際特許分類 (IPC))		
IntCl <sup>7</sup> G06F 9/30, G06F 9/38		
B. 調査を行った分野		
調査を行った最小限資料 (国際特許分類 (IPC))		
IntCl <sup>7</sup> G06F 9/30 - 9/42		
最小限資料以外の資料で調査を行った分野に含まれるもの		
日本国実用新案広報 1922-1996年 日本国公開実用新案広報 1971-2000年 日本国登録実用新案広報 1994-2000年 日本国実用新案登録広報 1996-2000年		
国際調査で使用了電子データベース (データベースの名称、調査に使用した用語)		
CSDB (日本国特許庁), 【後続/CW* [命令/CW+準備/CW+先行/CW]】 特許ファイル (PATOLIS), 【AP=アイビーフレックス株式会社】		
C. 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
X	JP, 7-253882, A (株式会社日立製作所) 3. 10月. 1995 (03. 10. 95) 全文, 第1-10図	14, 16, 19, 31, 33
A	全文, 第1-10図  (ファミリーなし)	17, 18, 20-30, 32, 34-39
<input checked="" type="checkbox"/> C欄の続きにも文献が列挙されている。 <input type="checkbox"/> パテントファミリーに関する別紙を参照。		
* 引用文献のカテゴリー 「A」 特に関連のある文献ではなく、一般的技術水準を示すもの 「E」 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの 「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す) 「O」 口頭による開示、使用、展示等に言及する文献 「P」 国際出願日前で、かつ優先権の主張の基礎となる出願日の後に公表された文献 「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの 「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの 「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの 「&」 同一パテントファミリー文献		
国際調査を完了した日	24. 10. 00	国際調査報告の発送日 31.10.00
国際調査機関の名称及びあて先	日本国特許庁 (ISA/JP) 郵便番号 100-8915 東京都千代田区霞が関三丁目4番3号	特許庁審査官 (権限のある職員) 伊知地 和之 電話番号 03-3581-1101 内線 3545

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
X	J P, 1-156824, A (株式会社日立製作所 外2名) 20. 6月. 1989 (20. 06. 89) 全文, 第1-11図	14, 16, 19, 31, 33
A	全文, 第1-11図  (ファミリーなし)	17, 18, 20-30, 32, 34-39
A	J P, 61-294548, A (株式会社東芝) 25. 12月. 1986 (25. 12. 86) 全文, 第1-6図 (ファミリーなし)	14, 16-39
A	J P, 2-183332, A (富士通株式会社) 17. 7月. 1990 (17. 07. 90) 全文, 第1-7図 (ファミリーなし)	14, 16-39

## 第Ⅰ欄 請求の範囲の一部の調査ができないときの意見 (第1ページの2の続き)

法第8条第3項(PCT17条(2)(a))の規定により、この国際調査報告は次の理由により請求の範囲の一部について作成しなかった。

1. ☐ 請求の範囲 \_\_\_\_\_ は、この国際調査機関が調査をすることを要しない対象に係るものである。つまり、
2. ☒ 請求の範囲 1-13, 15 は、有意義な国際調査をすることができる程度まで所定の要件を満たしていない国際出願の部分に係るものである。つまり、
  1. 請求の範囲1-13は、制御プログラム製品であるが、「製品」という技術範囲の明確でない用語を用いているために、請求の範囲1-13に係る発明を明確に把握することができない。
3. ☐ 請求の範囲 \_\_\_\_\_ は、従属請求の範囲であってPCT規則6.4(a)の第2文及び第3文の規定に従って記載されていない。

## 第Ⅱ欄 発明の単一性が欠如しているときの意見 (第1ページの3の続き)

次に述べるようにこの国際出願に二以上の発明があるとこの国際調査機関は認めた。

1. ☐ 出願人が必要な追加調査手数料をすべて期間内に納付したので、この国際調査報告は、すべての調査可能な請求の範囲について作成した。
2. ☐ 追加調査手数料を要求するまでもなく、すべての調査可能な請求の範囲について調査することができたので、追加調査手数料の納付を求めなかった。
3. ☐ 出願人が必要な追加調査手数料を一部のみしか期間内に納付しなかったため、この国際調査報告は、手数料の納付のあった次の請求の範囲のみについて作成した。
4. ☐ 出願人が必要な追加調査手数料を期間内に納付しなかったため、この国際調査報告は、請求の範囲の最初に記載されている発明に係る次の請求の範囲について作成した。

## 追加調査手数料の異議の申立てに関する注意

- ☐ 追加調査手数料の納付と共に出願人から異議申立てがあった。
- ☐ 追加調査手数料の納付と共に出願人から異議申立てがなかった。

## 第 I 欄 2. の続き

2. 請求の範囲 15 は、制御プログラムが埋め込まれている伝送媒体であるが、そもそも伝送媒体とプログラムの関係を何ら規定するものではなく、伝送しているプログラムでは物としての伝送媒体は特定されないため、請求の範囲 15 に係る発明を明確に把握できない。